

Coventry University



DOCTOR OF PHILOSOPHY

An Adaptive Fuzzy Based Recommender System For Enterprise Search

Alhabashneh, Obada

Award date:
2015

Awarding institution:
Coventry University

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of this thesis for personal non-commercial research or study
- This thesis cannot be reproduced or quoted extensively from without first obtaining permission from the copyright holder(s)
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 05. May. 2023

An Adaptive Fuzzy Based Recommender System For Enterprise Search

By

Obada Y. A. ALHABASHNEH

PhD

MAY 2015

*A thesis submitted in partial fulfilment of the University's requirements for the
Degree of Doctor of Philosophy*

ABSTRACT

This thesis discusses relevance feedback including implicit parameters, explicit parameters and user query and how they could be used to build a recommender system to enhance the search performance in the enterprise. It presents an approach for the development of an adaptive fuzzy logic based recommender system for enterprise search. The system is designed to recommend documents and people based on the user query in a task specific search environment. The proposed approach provides a new mechanism for constructing and integrating a task, user and document profiles into a unified index thorough the use of relevance feedback and fuzzy rule based summarisation. The three profiles are fuzzy based and are created using the captured relevance feedback. In the task profile, each task was modelled as a sequence of weighted terms which were used by the users to complete the task. In the user profile, the user was modelled as a sequence of weighted terms which were used to search for the required information. In the document profile the document was modelled as a group of weighted terms which were used by the users to retrieve the document. Fuzzy sets and rules were used to calculate the term weight based on the term frequency in the user queries. An empirical research was carried out to capture the relevance feedback from 35 users on 20 predefined simulated enterprise search tasks and to investigate the correlation between the implicit and explicit relevance feedback. Based on the results, an adaptive linear predictive model was developed to estimate the document relevancy from the implicit feedback parameters. The predicted document relevancy was then used to train the fuzzy system which created and integrated the three profiles, as briefly described above.

The captured data set was used to develop and train the fuzzy system. The proposed system achieved 89% accuracy performance classifying the relevant documents. With regard to the implementation, Apache Solr, Apache Tikka, Oracle 11g and Java were used to develop a prototype system. The overall retrieval accuracy performance of the proposed system was tested by carrying out a comparative retrieval accuracy performance evaluation based on Precision (P), Recall (R) and ranking analysis. The values of P and R of the proposed system were compared with two other systems being the standard inverted index based Solr system and the semantic indexing based lucid system. The proposed system enhanced the value of P significantly where the average of P value has been increased from 0.00428 to 0.064 as compared with the

standard Solr and from 0.0298 to 0.064 compared with Lucid. In other words, the proposed system has managed to decrease the number of irrelevant documents in the search result which means that the ability of the system to show the relevant document has been enhanced. The proposed system has also enhanced the value of R . The average value of R has been increased significantly (doubling) from 0.436 to 0.828 as compared with the standard Solr and from 0.76804 to 0.828 as compared with Lucid. This means that the ability of the system to retrieve the relevant document has also been enhanced. Furthermore the ability of the system to rank higher the relevant documents has been improved as compared with the other two systems.

ACKNOWLEDGMENT

I would like to express my deep gratitude to my director of studies Dr. Rahat Iqbal for all of his time, support and courage. He was always there for me and made sure that I got back on track whenever it was needed. I also need to thank Dr. Faiyaz Doctor and to say to him that his support and help will be always appreciated.

Big thanks as well to Dr. Saad Amin and Professor Anne James for their support. I would like to express my appreciation to volunteers who participated in the user study and gave their time and effort to make this research successful. Finally, I would like to gift this work to my mother and father who didn't spare any time and effort to make me always happy and successful.

TABLE OF CONTENTS

1. CHAPTER 1 INTRODUCTION	9
1.1. INTRODUCTION.....	9
1.2. MOTIVATION	10
1.3. PROBLEM STATEMENT	12
1.4. AIM & OBJECTIVES	12
1.5. RESEARCH QUESTIONS.....	12
1.6. RESEARCH SCOPE	13
1.7. RESEARCH METHODOLOGY	14
1.8. RESEARCH CONTRIBUTION	17
1.9. STRUCTURE OF THE THESIS	17
2. CHAPTER 2 BACKGROUND	20
2.1. INTRODUCTION.....	20
2.2. ENTERPRISE SEARCH	20
2.2.1. ENTERPRISE SEARCH VERSUS INTERNET SEARCH.....	21
2.2.2. EXPERT SEARCH IN ENTERPRISE.....	23
2.2.3. KEY RESEARCH PROBLEMS IN ENTERPRISE SEARCH.....	23
2.3. RECOMMENDER SYSTEMS.....	25
2.3.1. CONTENT-BASED RECOMMENDATION SYSTEMS	25
2.3.2. COLLABORATIVE FILTERING (CF) RECOMMENDATION SYSTEMS.....	25
2.4. CONCLUSION	27
3. CHAPTER 3 LITERATURE REVIEW	29
3.1. INTRODUCTION.....	29
3.2. ENTERPRISE SEARCH	30
3.2.1. EXPERT SEARCH AND RECOMMENDATION.....	32
3.3. RECOMMENDER SYSTEMS.....	34
3.4. USER PROFILE	38
3.4.1. USER PROFILE CONTENTS	39
3.4.2. GROUP PROFILES.....	40
3.5. RELEVANCE FEEDBACK	41
3.6. MACHINE LEARNING FOR RECOMMENDER SYSTEM	46
3.6.1. FUZZY LOGIC.....	51

3.7. CONCLUSION	55
4. CHAPTER 4 USER STUDY	57
4.1. INTRODUCTION.....	57
4.2. USER STUDY	57
4.2.1. PARTICIPANTS.....	58
4.2.2. DATASET (TREC Enterprise Trak 2007).....	58
4.2.3. SEARCH TASKS	60
4.3. USER STUDY EXPERIMENTAL SETUP	62
4.4. DATA COLLECTION.....	69
4.5. CONCLUSION	70
5. CHAPTER 5: PROPOSED APPROACH	72
5.1. INTRODUCTION.....	72
5.2. PROPOSED APPROACH	73
5.2.1. PHASE 1: RELEVANCE FEEDBACK COLLECTION	75
5.2.2. PHASE 2: DOCUMENT RELEVANCE PREDICTION.....	76
5.2.3. PHASE 3: FUZZY BASED TASK, USER AND DOCUMENT PROFILING.....	79
5.2.4. PHASE 4: FUZZY COMBINED WEIGHT CALCULATION & UNIFIED TERM WEIGHT INDEX (UTWI) CREATION	86
5.2.5. PHASE 5: RECOMMENDATION OF DOCUMENTS AND PEOPLE (EXPERTS).....	91
5.2.6. PHASE 6: RECOMMENDATION PRESENTATION.....	92
5.3. IMPLEMENTATION	92
5.3.1. DOCUMENT RELEVANCE PREDICTION COMPONENT.....	93
5.3.2. FUZZY PROFILES CREATING COMPONENT	94
5.3.3. UNIFIED TERM WEIGHT INDEX (UTWI) CREATING COMPONENT.....	96
5.3.4. RECOMMENDATIONS CREATING COMPONENT	99
5.4. CONCLUSION	100
6. CHAPTER 6: RESULTS AND EVALUATION	101
6.1. INTRODUCTION.....	101
6.2. LINEAR PREDICTIVE MODEL VALIDATION.....	102
6.3. RULE BASED SUMMARISATION VALIDATION USING K-FOLD.....	103
6.4. EVALUATION USING PRECISION, RECALL AND RANKING ANALYSIS.....	107
6.4.1. PRECISION AND RECALL ANALYSIS	107
6.4.2. COMPARATIVE DOCUMENT RANKING ANALYSIS	110
6.5. CONCLUSION	111

7. CHAPTER 7: CONCLUSION	113
7.1. INTRODUCTION.....	113
7.2. RESEARCH SUMMARY	113
7.3. CONTRIBUTION.....	114
7.4. RESEARCH LIMITATIONS	117
7.5. FUTURE WORK	118
8. REFERENCES	120
9. APPENDICES	134
9.1. APPENDEX1: INFORMATION SHEET & SEARCH TASK	134
9.2. APPENDIX 2: THE CONSENT FORM	139
9.3. APPENIX 3: USER GUIDE	140
9.4. APPENIX 4: ETHICAL APPROVAL.....	Error! Bookmark not defined.

LIST OF FIGURES

FIGURE 1.1 : RESEARCH METHODOLOGY	14
FIGURE 3.1: USER-ITEM SIMILARITY MATRIX (RICCI ET AL. 2011).....	35
FIGURE 3.2 : OARD AND KIM (2001) CLASSIFICATION FOR POTENTLY OBSERVABLE BEHAVIOUR (IMPLICIT FEEDBACK PARAMETERS)	43
FIGURE 3.3: KELLY AND TEEVAN (2003) EXTENDED CLASSIFICATION FOR IMPLICIT FEEDBACK PARAMETERS	44
FIGURE 3.4: CRISP SET	52
FIGURE 3.5: FUZZY SETS VL, L, M, H AND VH.....	53
FIGURE 3.6: FUZZY SET AND CRISP SET.....	53
FIGURE 3.7: MEMBER FUNCTION SHAPES B (UNIVERSITY OF STRATHCLYDE 2015).....	54
FIGURE 3.8: FUZZY SETS A AND B (UNIVERSITY OF STRATHCLYDE 2015).....	54
FIGURE 3.9: FUZZY OPERATION EXAMPLE (UNIVERSITY OF STRATHCLYDE 2015).....	55
FIGURE 4.1: SAMPLE OF THE LABELLED DATA	60
FIGURE 4.2: SAMPLE OF THE DEVELOPED TASKS.....	61
FIGURE 4.3: USER STUDY EXPERIMENTAL SET UP	62
FIGURE 4.4: LOGIN SCREEN.....	63
FIGURE 4.5: SEARCH TASKS SCREEN.....	63
FIGURE 4.6: SEARCH SCREEN	64
FIGURE 4.7 : DOCUMENT SCREEN.....	64
FIGURE 4.8: EXPLICIT FEEDBACK SCREEN	65
FIGURE 4.9: RELEVANCE FEEDBACK COLLECTION.....	66
FIGURE 5.1 : PROPOSED APPROACH.....	74
FIGURE 5.2 : RELEVANCE FEEDBACK COLLECTION.....	76
FIGURE 5.3 : PREDICTORS & TARGET	77
FIGURE 5.4 :FUZZY BASED TASK, USER AND DOCUMENT PROFILING.....	80
FIGURE 5.5: FUZZY SETS FOR INPUT VARIABLES.....	82

FIGURE 5.6 : WT CALCULATION FUZZY RULES	82
FIGURE 5.7 : FUZZY SETS FOR INPUT AND OUTPUT VARIABLES	87
FIGURE 5.8: RECOMMENDER SYSTEM USER INTERFACE.....	92
FIGURE 5.9: PROPOSED SYSTEM ARCHITECTURE	93
FIGURE 5.10 : TERMS FREQUENCIES VIEWS.....	94
FIGURE 5.11: FUZZY CONTROLLER (A): PROFILE TERM WEIGHT.....	95
FIGURE 5.12: PROFILE TERM WEIGHT FUZZY CONTROLLER SIMULATION.....	95
FIGURE 5.13 : PROFILES DATABASE TABLES.....	96
FIGURE 5.14 : TERM VISIT WEIGHTS	96
FIGURE 5.15 : BEST FUZZY RULES	97
FIGURE 5.16: FUZZY CONTROLLER (B): UNIFIED TERM WEIGHT	98
FIGURE 5.17: UNIFIED TERM WEIGHT FUZZY CONTROLLER.....	98
FIGURE 5.18: UTWI DATABASE TABLE. TASK_USER DATABASE VIEW AND TASK DOCUMENT DATABASE VIEW	99
FIGURE 5.19: SELECT STATEMENT FOR RECOMMENDED USER (PEOPLE) LIST .	99
FIGURE 6.1 :PIVOT OF THE PREDICTED VALUE & ACTUAL VALUE.....	103
FIGURE 6.2:PRECISION (P) AND RECALL (R) FOR: STANDARD VECTOR SPACE SEARCH SYSTEM (STD SOLR), SEMANTIC BASED SEARCH SYSTEM (LUCID SOLR) AND THE PROPOSED RECOMMENDER SYSTEM.....	109
FIGURE 6.3: COMPARED DOCUMENT FREQUENCIES FOR RANK CATEGORIES	111

LIST OF TABLES

TABLE 3.1 : FUZZY RULES EXAMPLE.....	55
TABLE 4.1:PARTICIPANTS' CHARACTERISTICS.	59
TABLE 4.2: RELEVANCE FEEDBACK PARAMETERS' DESCRIPTION.....	69
TABLE 4.3: SAMPLE OF RELEVANCE FEEDBACK DATA	70
TABLE 5.1 : CORRELATION ANALYSIS	78
TABLE 5.2 : COEFFICIENTS FOR THE TARGET EXPLICIT RELEVANCE FEEDBACK	79
TABLE 5.3 : SAMPLE OF USER PROFILE.....	83
TABLE 5.4 : SAMPLE TASK PROFILE.....	84
TABLE 5.5 : SAMPLE OF THE DOCUMENT PROFILE.....	86
TABLE 5.6 : SAMPLE OF THE EXTRACTED FUZZY RULES	88
TABLE 6.1:SUM SQUARES FOR THE LINEAR MODEL	102
TABLE 6.2: SUMMARIZED WEIGHTED FUZZY RULES FOR K=1	104
TABLE 6.3: SAMPLE OF SUMMARISED FUZZY RULES ACCURACY	105
TABLE 6.4: K-FOLD ACCURACY	106
TABLE 6.5: SUMMARIZED WEIGHTED FUZZY RULES FOR K=4.....	106
TABLE 6.6: PRECISION (P) AND RECALL (R) FOR: STANDARD VECTOR SPACE SEARCH SYSTEM (STD SOLR), SEMANTIC BASED SEARCH SYSTEM (LUCID SOLR) AND THE PROPOSED RECOMMENDER SYSTEM.....	108
TABLE 6.7:COMPARED DOCUMENT FREQUENCIES FOR RANK CATEGORIES ..	110

1. CHAPTER 1 INTRODUCTION

1.1. INTRODUCTION

Information has become one of the most important organisational needs in order to survive in the highly competitive business environment that we witness today. Finding the right information when it is required is crucial and selecting the wrong information can impact both the business processes and the decision making of the enterprise. There has been reported a noticeable dissatisfaction among information workers with the retrieval performance of the current enterprise search tools in their organisations. The poor retrieval performance, along with growth of the information available for enterprises, overloads information workers with a lot of irrelevant information which impacts the efficiency of the organisation. There are number of structural differences between the enterprise search and the Web. For example, the anchor texts which link web documents together and are used as a base for the PageRank algorithm in the Web search are not found in enterprise documents. Secondly the heterogeneity of documents means that different algorithms are required to process them, different ranking mechanisms are required to prioritise them and they need different levels of access control to protect them.

Recommender systems could be used to enhance the search result accuracy in the enterprise and minimise this information overloading. They can be developed using relevance feedback based approaches (Ricci et al. 2011) which determine the relevance of a particular piece of information (document) to the user and how its content can be reused in order to find documents that are similar. The use of relevance feedback increases the chance that similar documents can be retrieved which may go some way to offset the lack of anchor texts as well as providing contextual information about the needs of the information worker.

The two most widely recognised techniques of relevance feedback are explicit and implicit (Amatriain et al. 2009; Anand, Kearney, Shapcott, 2007; Hu, Koren, Volinsky 2008). In explicit feedback, users mark the documents explicitly as relevant or not relevant. In implicit feedback, the relevance is estimated by observing the behaviour of search users when processing information and then collecting

relevance parameters. The relevance parameters include reading time, click count, text section, etc. Profiles of the user can be developed using the relevance feedback approaches. One of the significant techniques used in recommender systems is user profiling, where such profiles contain browsing history, tasks performed, preferences and interests (Schiaffino and Amandi, 2009; Brusilovsky and Millán, 2007). However, relevance feedback involves a high level of uncertainty due to the inconsistency in user behaviour and subjectivity in their assessment of relevancy (Kearney, Shapcott, 2007; Hu, Koren, Volinsky 2008). Therefore, handling such uncertainty is crucial to achieve better performance. Fuzzy logic has been used to deal with uncertainty in different application domains ranging from the controllers systems (Skalistis, Petrovic, Shaikh, 2013) to information retrieval (IR) Eckhardt (2012). It can be used to enhance the search result accuracy by handling the uncertainty and ambiguity in user data as fuzzy sets provide an expressive method for user judgment modelling and fuzzy rules provide an interpretable method of classifying the most relevant results.

This thesis presents an approach for the development of a fuzzy recommender system to enhance the process for searching for documents containing the relevant information and also searching for people by identifying the “experts” in a particular topic area in the enterprise. This approach provides a new mechanism for constructing and integrating profiles for the task, user and document, into a unified index by the use of relevance feedback and fuzzy rule based summarisation.

The rest of the chapter is organised as follows: Section 1.2 discusses the motivation of the research. Section 1.3 discusses the research problem. Section 1.4 discusses the aim and the objectives of the research. Section 1.5 discusses the research questions. Section 1.6 discusses the scope of the research. Section 1.7 discusses the research methodology. Section 1.8 discusses the structure of the rest of the thesis.

1.2. MOTIVATION

Information has become one of the important resources of the organisation that is essential to survive in the highly competitive business environment that we witness today. According to the European Commission report 2013 (White et al. 2013), organisations lose 14% of their potential revenue every year as a cost of the poor quality of retrieved information. Hawking (2010) argued that from the results of the study of the Butler Group (2006), the cost of finding the required information equated to 10% of the salary cost of the organisation.. In the ICD report (2005), it was found

that employees spend 20% of their time on average, searching for information they could not use, which meant that an organisation with 1,000 employees wasted \$2.5 million annually because of poor search capability.

Other available evidence also shows that enterprise search tools are inefficient and unable to meet the expectation of the users and clients. The recent FindWise survey on information findability (2013) showed that only 9% of information searchers believed that it was easy to find the required information within an organisation while 63% believed it was hard. It also showed that 60% were dissatisfied with the search tools provided (Norling, 2013). Middle managers believe that more than 50% of the information retrieved by the search tools was irrelevant. On the other hand, there has been a high growth of information as a resource of the enterprise. According to the European Commission report 2013 (White et al. 2013), the amount of information collected and managed by European organisation has increased by 86% since 2007.

Comparing the enterprise search with robust web search tools could raise the question: why not apply these robust tools and methods to achieve better retrieval performance in the enterprise?. Web search tools are described as inefficient for the enterprise search because there are structural differences in the nature of the information on the Web compared to the enterprise information (Broder and Ciccolo, 2004; Mukherjee and Mao, 2004). For example, 80% of enterprise information consists of non-web documents, which means the documents are not connected to each other with hyperlinks. This limits the efficiency of the most common web search ranking algorithms such as PageRank.

The importance of the information, poor retrieval performance of the current enterprise search tools and the high level of growth of digital information available for enterprises, has created an urgent need for intelligent approaches to enhance the search quality for information in the enterprise. The total enterprise search market in Europe has reached €500 million by 2013 while the world market has reached €2 billion (White et al. 2013). This high growth in the enterprise search market reflects the need for organisations to have efficient enterprise search tools. Grefenstette (2009) stated that the growth average of the enterprise search industry is around 20% per year which indicates the need for a robust enterprise search engine to meet the information needs of the enterprise.

Although, enterprise search has received increasing attention from vendors, organisations and the research community, the amount of research is relatively limited

and the outputs from studies of this research area are still lacking (Pavel et al. 2010, Grefenstette 2009).

1.3. PROBLEM STATEMENT

Poor retrieval performance of the current enterprise search tools and the large amount of searchable information in enterprises has caused information overload for users searching for information. Time and effort is wasted searching for relevant information or having to use irrelevant information which affects the quality of services and decision making within the organisation. This research is an attempt to address the problem of information overload by improving the retrieval accuracy of enterprise search. This will be achieved by developing an intelligent recommender system which is able to filter out the irrelevant search result and display those documents and experts which are relevant to the user query. A list of experts will include those people who are most likely to have the required knowledge of the query topic and have searched/read those documents before.

1.4. AIM & OBJECTIVES

This research aims to enhance the retrieval accuracy in the enterprise search by proposing an adaptive intelligent approach for recommender systems based on relevance feedback. The aim will be achieved through the fulfillment of the following objectives:

- To explore the current methods, techniques, tools and issues in enterprise search.
- To investigate the relevance feedback approaches, including: implicit parameters, explicit parameters and query.
- To investigate the relationship between implicit and explicit feedback parameters in order to identify the most reflective parameters for the user interest.
- To propose an intelligent and adaptive approach for recommender systems in order to improve the accuracy of the enterprise search result.
- To evaluate the accuracy of the proposed approach by identifying the relevant documents retrieved from a user query.

1.5. RESEARCH QUESTIONS

This research is carried out to answer the following main question:

How could an intelligent recommendation improve the retrieval accuracy in

enterprise search and in turn help to address the information overloading problem in the enterprise. This broad research question could be answered through answering the following sub questions:

- What are the main challenges and issues that limit the retrieval accuracy of the enterprise search?
- What are implicit and explicit feedback parameters? Is there any relationship between them and how could they be used to enhance the retrieval accuracy of the enterprise search?
- How can the search result accuracy be enhanced in the enterprise environment by using user feedback?

1.6. RESEARCH SCOPE

The scope of the research project is limited as follows:

- Only open source and freely available technologies are used.
- The main focus of the research is the search result accuracy and not other aspects of the performance such as response time, scalability or complexity.
- The proposed system is designed to work as an upper layer on the top of the search facility in the enterprise to filter out the irrelevant documents of the search results and does not deal with aspects of the indexing process.
- Information overload occurs when the quantity of information to be processed is more than the individual can process in the time available for processing (Jackson 2001; Ruff 2002). In the context of the search process, it occurs when the number of items returned by the search engine are large and not relevant to the user query. Information filtering is one of the common approaches to address this problem as it aims to improve retrieval accuracy by enhancing the value of both precision and recall minimising the number of irrelevant documents retrieved.
- It is assumed that the search tasks are related to the user role in the organization. They are predefined and provided according to the taxonomy of the enterprise.
- Due to the data access limitation, only document search and people search based on the search history of users will be considered.
- The results are limited to documents and user queries which are written in the English language.
- Results and findings will be limited to cases where the user behaviour is

consistent.

1.7. RESEARCH METHODOLOGY

In order to achieve the aim and objectives of the research a multi-step methodology will be applied. As shown in *Fig (1.1)* The methodology consists of a number of steps which include problem identification and definition, proposing the approach to address the research problem, carrying out a user study to capture data for implementing the approach, training and validating the proposed approach models, evaluating the retrieval accuracy of the proposed approach, and finally drawing up the conclusions and recommendations. During the research process an on-going literature review will be carried out in order to understand the problem domain, gaps in the knowledge and the perceived limitations of the existing approaches so that a suitable and effective approach can be developed.

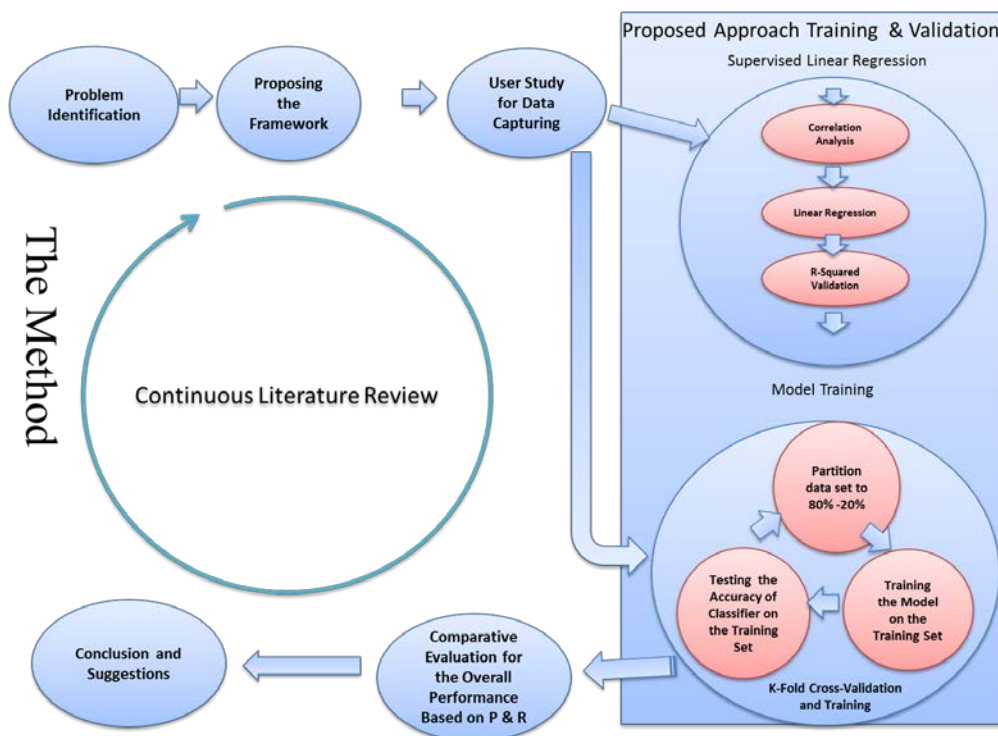


FIGURE 1.1 : RESEARCH METHODOLOGY

Step 1, Problem Identification: This step surveys the literature to provide a better understanding of the problem domain and context. The research problem is identified and defined in order to set the research aim and objectives clearly.

Step 2, Uroposed approach: This step develops the approach to address the research problem which includes the component identification, design and interfaces.

Step 3, User study: This step will be carried out to create the dataset which is required to implement and train the proposed approach. The data set will be captured from users using the controlled observation technique (Magnusson et al. 2009; Gulliksen et al. 2003)

Step 4, Training and validation: This step will train and validate the models in the proposed approach. The approach models will be trained and validated on multiple passes. The process will apply two supervised machine learning tasks, specifically, regression and classification. Machine learning uses computational methods to make the computer learn from past experience in order to improve future performance (Cintra 2005; Alpaydin 2014).

In general, machine learning can be categorised into supervised, semi supervised, unsupervised and reinforced. In supervised learning the predicted output data for given input data is provided by a supervisor (i.e. the data is labelled) and this is then used for training the model to deal with future similar data. Regression and classification are commonly used tasks in this type of learning (Alpaydin 2014). Semi-supervised learning uses a mixture of labelled and unlabelled data to train the model. The labelled data is used to create the prediction model, which is then used to produce predictions for a subset of the unlabelled data. The resulting predictions are then used to label the data which will be used for the future training of the model. In unsupervised learning, there is no labelled data at all and the system is trained to group and cluster the inputs rather than just making predictions of the output. The common term for this type of learning is data clustering. In reinforcement learning there is no labelled output and even the inputs are not clearly predefined.

In order to train models for the proposed approach, supervised learning which includes regression and classification will be used. Regression will be used to investigate the relationships between the implicit user feedback and the relevance of the retrieved document. Classification will be used to estimate the weights of the user query terms for each of the user, task and document profiles. These profiles will then be combined into a unified index. Knowledge extraction and compression will also be used in order to extract the classifying rules.

Regression is supervised learning in which a regression model is trained on a labelled data set to predict the value of an output variable from the values of input variables. The regression model consists of the given input variables with their associated coefficients, which represent the influence of each input variable in predicting the value of the output variable value. Regression analysis will be discussed in more details in *Chapter 4*.

Classification is a supervised machine learning task, in which the system is trained to classify the data into two or more categories based on a set of defined rules. Classification is useful in data discrimination and prediction. In discrimination the classification of data takes place, whereas in prediction classifying rules are used to predict the output value of new input data. There are different of classification methods such as decision trees, Bayesian method and artificial neural networks (Alpaydin 2014). However, a comparison between these methods will be carried out in order to select the best fit classification method for the research.

Validation

In this research two validation methods will be applied, R-squared (R^2) and ***K-fold*** cross validation. R-squared is a common validation method for the regression model which is based on the squared differences between the predicted values and their averages; and between the actual values and their average (Arlot and Celisse, 2010). This method is described in detail in *Chapter 6*. K-fold cross validation will be used to validate the accuracy of the classifiers. This method will be discussed in detail in *Chapter 6*.

Step 5: Overall retrieval accuracy comparative evaluation: In this step the overall retrieval accuracy of the proposed approach is evaluated. The evaluation will be based on the well-known retrieval accuracy matrices, Precision (P) and Recall (R) (Kelly, 2008). The precision and recall of the proposed approach will be compared with both the standard inverted index based enterprise search (standard Solr) and semantic indexing based enterprise search tool (Lucid) for the same data set. In addition a document ranking performance evaluation will be carried out in order to assess the ability of the system to show the relevant documents at the top of the search result.

Step 6: Drawing conclusions and future work identification: Based on the knowledge gained throughout the research process and results, the main conclusions will be drawn to summarise the research. In addition the future work will be identified

in order to extend the research in the future and also to direct the work of other researchers in the subject area.

1.8. RESEARCH CONTRIBUTION

The main contribution that this research has made to existing knowledge was the development of an adaptive integrated fuzzy approach for a recommender system, to be used in enterprise search. The recommender system was used to recommend relevant documents based on relevance feedback. In addition, the system also recommended people who have expertise in the search area. The contribution this research made to the existing knowledge can be summarised as follows:

- The empirical research carried out as part of this thesis has clearly found significant co-relation between implicit parameters (i.e. time on page, mouse movements, and mouse clicks) and the explicit document relevancy.
- An adaptive linear predictive model was developed to estimate the document relevance from the implicit feedback parameters.
- A new approach for profiling was proposed. The approach extended the method proposed in (Li & Kim 2004) to include task, user and document profiles, rather than only creating a user profile.
- An adaptive fuzzy mechanism was developed to integrate the three profiles into one index that contained a unified term weight for each occurrence of the term in the user queries.
- As a result of the research experiments, the labelled data of the well-known enterprise search test collection ‘TREC Enterprise Track 2007’ was extended to include more user queries for the topics provided and relevance feedback (implicit and explicit) on the created queries.

The research contribution will be discussed in more details and in relation to the research objectives in *Chapter 7*.

1.9. STRUCTURE OF THE THESIS

The thesis will be constructed as follows:

- **CHAPTER 1: INTRODUCTION**

This chapter presents an overview of the current research. It highlights the research importance and also gives a brief background about the problem domain. The aim and

objectives of the research, research questions, scope, problem statement and the structure of the thesis are discussed.

- **CHAPTER 2: BACKGROUND**

This chapter provides the background and context of the research problem by giving an introduction to enterprise search, recommender systems and fuzzy logic, the essential concepts of this research. The chapter starts by defining the enterprise search and how it differs from the web search, while considering the issues which have been experienced with the enterprise search. The definition, structure and the main approaches for recommender systems are then discussed together with an introduction to fuzzy logic systems and their main components (e.g. fuzzy sets, member functions and fuzzy rules).

- **CHAPTER 3: LITERATURE REVIEW**

This chapter presents a review of the related literature. The information presented in this chapter includes: enterprise search; recommender systems and their application in the enterprise search; relevance feedback and its application in the enterprise search; intelligent approaches for recommender systems; and fuzzy logic together with its application for recommender systems.

- **CHAPTER 4: USER STUDY**

This chapter discusses the user study which was conducted as a part of the research to capture the relevance feedback from 35 search users, based on an enterprise document test collection. The relevance feedback was captured in order to maintain an adequate amount of data for the profiling process used by the proposed approach. It also provided the means to conduct empirical research, to gain a better understanding of the nature of the relationship between implicit feedback and the relevance level of the retrieved document within the context of the enterprise.

- **CHAPTER 5: THE PROPOSED APPROACH**

This chapter discusses the proposed approach and its various phases, which include:

- Feedback collection from users of the search including what data was captured and how it was captured.
- Document relevancy prediction based on the developed linear predictive model and how the model was developed using correlation and regression analysis.

- Fuzzy logic based Task, User and document profiling process and how these profiles were created and structured.
- Fuzzy combined weight calculation and Unified Term Weight Index (UTWI) creation, based on the fuzzy rules summarization approach.
- The iterated training and validation process for applying the rules summarization approach, in order to extract the best set of rules for combined weight calculation.
- Recommendations creation based on the relevance between the user query and the relevant search task, user and document.
- Recommendations presentation to the user through a web based user interface.
- The implementation of the proposed approach

• **CHAPTER 6: THE EXPERIMENTS AND RESULTS**

This chapter discusses the evaluation methods and results for the proposed approach. The proposed approach was evaluated at two levels: the validation of the accuracy of the component and the overall retrieval performance. The linear predictive model and fuzzy system was built based on the summarised fuzzy rules. The linear predictive model was validated using the R-squared method and the fuzzy system was validated using K-Fold method. The overall retrieval accuracy of the proposed system was tested by carrying out a comparative retrieval accuracy evaluation based on Precision (P) and Recall (R) in which the values of P and R were compared with two information retrieval systems.

• **CHAPTER 7: CONCLUSIONS**

This chapter discusses the main contributions and outcomes of the research and how the research objectives were achieved by reference to the relevant chapters of the thesis. The research limitations and future work are also discussed in this chapter.

2. CHAPTER 2 BACKGROUND

2.1. INTRODUCTION

The previous chapter discussed the aim, objectives, motivation and scope of the research. A systematic methodology was also defined in order to address the problem and to achieve the objectives of this research. As briefly discussed in the previous chapter, the highly competitive business environment has increased the importance of information as one of the organisational resources. The large amount of information within enterprises and on the internet and internal servers created a critical need for an effective information retrieval system. The availability of the correct information is crucial for timely decision making. Current enterprise search tools are not robust enough to meet the user information needs.

This chapter sheds more light on the problems of the enterprise search and its importance to the organisation and highlights the main differences between the Internet and Intranet search. It also provides background information for recommender systems

The rest of the chapter is organised as follows: Section 2.2 discusses the enterprise search and considers the advantages which this type of search has over the more traditional web search in respect of retrieval performance. Section 2.3 discusses the main approaches used for recommender systems. Section 2.4 concludes the chapter.

2.2. ENTERPRISE SEARCH

Many enterprises have a rich and diverse collection of various information resources. Such resources can be divided into structured and unstructured information. Structured information is encoded into databases while unstructured information is encoded into documents. The retrieval of structured information has been well investigated in the literature (Mangold et al. 2006) and several commercial search tools or products are available in the market in the form of the traditional database engines (e.g., Oracle, Microsoft SQL server, MySQL). However, retrieval of unstructured information is still a challenging task due to the lack of anchor text (e.g., hyperlinks), heterogeneous format of documents and other problems.

Hawking (2010) defined the enterprise search as "the application of information retrieval technology to information finding within organisations". This information finding includes various information sources including digital documents, emails, database records and webpages, which are owned by the organisation.

Enterprise Search Engines (ESEs) are still not advanced and mature enough to provide the required high quality results (Hawking, 2004; IDC, 2004; Dmitriev et al.2010; White et al. 2013). Despite the fact that there are various companies that provide enterprise search solutions such as Google, Verity, IBM and Panoptic, little research work has been done in this area of enterprise research. (Dmitriev et al. 2006; Owens, 2008; Alhabashneh et al. 2012; Hawking 2004).

2.2.1. ENTERPRISE SEARCH VERSUS INTERNET SEARCH

Information retrieval is a complex and cognitively demanding task. Particularly, searching for information in an enterprise is challenging, as accessing information from various diverse resources is difficult and even sometimes impossible. Furthermore, the information resources might be part of different systems or subsystems imposing further administrative, privacy and security policies within the context of an enterprise (Broder and Ciccolo, 2004; Mukherjee and Mao, 2004). On the other hand, the information on the web, although changing dynamically, is easily publicly accessible.

Information in the enterprise is multi-dimensional, that is it can be structured, semi-structured or unstructured. Furthermore, information could be written in different languages and distributed on different platforms and locations. The metadata about the documents could be limited as well and the documents themselves could be formatted differently (e.g. Word documents, PowerPoint presentations and Excel spreadsheets).

Hawking (2010), argued that the need for a federate search in which the user is provided with a single list as a search result, with retrieved information ranked by importance, adds more challenge for the enterprise search tools developer. Creating such a list over a variety of document types, access rights, repositories and contexts is very difficult. The user is unaware of the complications beyond the search screen and requires information retrieval in the enterprise to be as easy and efficient as in the web search.

There is a wide range of commercial ESE products that are available for the customer from many vendors, such as; Google, Verity, IBM, Oracle, Microsoft and Panoptic. Unfortunately, none of the existing enterprise search products provide a full solution for the enterprise information needs. (Dmitriev et al. 2006; Owens 2008; Alhabashneh et al. 2012; Hawking 2004). Hawking and Zobel (2007) found that limitations within the company and improper implementation policies made the effectiveness of metadata mark-up in enterprise search invaluable, in spite of committed resources. Even though there is a transfer of technology from web search to enterprise search, there are vast differences between them that have been explained in this chapter. A major difference is that no organisation rewards spam information.

The definition of an answer to a query varies on the internet as compared to that on the intranet. The internet provides all possible answers to the query and the user selects the best or the most relevant of them. On the contrary, the enterprise is governed by the notion of finding the right answer, which may differ from the best answer on the internet. Arriving at the right solution to a problem is indeed a different task than looking for the best solution (Fagin et al. 2003; Raghavan 2001; Hawking 2004).

The social forces driving the content of Internet and intranet differ in many ways. The Internet is a reflection of the collaborative opinions of a number of authors who exercise their freedom to publish content, as opposed to an intranet that serves a particular organisation and should only reflect the viewpoint of people in that organisation. Internet content is focused on information dissemination, rather than building traffic or targeting any number of viewers. Content creation is not sought as an incentive-building activity and the right to publish content is not granted to everyone in the organisation. Information gathered from different repositories (e-mail systems and content management systems) are generally not cross-referenced through hyperlinks. Therefore, there is a difference between the amount of linked pages or documents on an intranet and on the Internet. For instance, on the Internet the powerfully linked sections (links connecting different pages) make up for 30 % of visited pages while this number is far less on corporate intranets (Hawking 2010).

PageRank and HITS techniques that are popular on the Internet are of little use on an intranet, thus demanding the employability of other methods to improve intranet search. Certain characteristics of enterprise content and processes have made the enterprise information retrieval (IR) systems differ from those of the Web, thus

causing differences in the way that enterprise search has evolved. (Fagin et al. 2003, Raghavan 2001, Hawking 2004).

2.2.2. EXPERT SEARCH IN ENTERPRISE

In large organisations, information retrieval is often accompanied by the need to search for other users/colleagues who possess knowledge of the topic (Hertzum and Pejtersen, 2000). Also, it has been seen that the relevant information is not present in electronic format or cannot be deciphered or converted into a written language. In such situations, taking help from other people becomes necessary (Craswell et al. 2001). There are experts who give logical and satisfying answers to particular queries, and may provide links to gather further information. Such experts are always in demand by event organizers who constantly look for consultants, analysts, and talent hunters to get their expertise from tackling client enquiries and keeping intact their client-base (Idinopulos and Kempler, 2006).

Though it may be difficult for an organisation to find such experts, identifying them via social media or professional networks is difficult in large firms, which may be located in different geographical areas.. Generally, to facilitate the search for people or departments with specialised knowledge and skills both within and outside the firm, a specialised expert search tool is required (Maybury, 2006). Recruitment costs are reduced and money saved if an expert can be found at a reasonable cost and convenient location in another organization. An expert finder tool powered by a text search engine requires a small set of queries by the user as input and produces an index listing all persons with the required skill/knowledge on the topic as output. This system ensures that the information is traceable by providing relevant documents (notes of documents written by the persons listed) evidencing the expertise of the listed individuals (Hawking, 2004).

2.2.3. KEY RESEARCH PROBLEMS IN ENTERPRISE SEARCH

The Enterprise Search Engine faces significant problems that limit its ability to rank heterogeneous documents, estimate non-web document importance and extract and utilise search context (Hawking 2010; Alhabashneh 2011; Owens 2008).

2.2.3.1. RANKING HETEROGENEOUS DOCUMENTS

Most enterprise documents are non-web documents, and furthermore they are heterogeneous (have different types, structure, purposes and nature). This makes webpage ranking techniques inefficient for Enterprise Search. For example, PowerPoint files consist of slides and each slide has a title and body. The title part, logically, should have a higher importance than the body part. On the other hand, Excel spreadsheets have a structure of columns and rows and always consist of numerical values with a limited text description as well as the column titles. The question now is: how could the same ranking algorithm or technique be applied to very different file types? It is obvious that the text based ranking methods are not effective in this case (Alhabashneh 2011; Hawking 2010; Dmitriev 2010; Owens 2008; Mangold 2006).

2.2.3.2. NON-WEB DOCUMENT IMPORTANCE ESTIMATION

The structure analysis showed that the Enterprise web does not follow the bow-tie structure of the WW-Web pages which makes the Page-Rank algorithm inefficient for the enterprise search. Most enterprise documents have no anchor texts, since the anchor texts are used by the web search engines as the means to calculate the document importance in the document ranking algorithm (Hawking 2010; Dmitriev 2010; Owens 2008). The lack of such texts make these algorithms inefficient in the enterprise search case (Alhabashneh 2011; Hawking 2010; Dmitriev 2010; Owens 2008; Mangold 2006).

2.2.3.3. EXTRACTING AND USING THE SEARCH CONTEXT

Search context is useful to disambiguate the short or ambiguous queries, since it adds more keywords to the user query, which makes it clearer to the search and easier to correctly rank the result list. The user profile (e.g. reading age, first language, interests, search history, user feedback etc.) can be accessed to give context to the search. The problem here, however, is how this content can be used effectively in the search. (Alhabashneh 2011; Hawking 2010; Dmitriev 2010; Owens 2008; Mangold 2006).

2.3. RECOMMENDER SYSTEMS

Recommender systems (RS) can be described as intelligent systems which are provided with the capability to suggest information to the users (Ricci et al. 2011; Burke, 2007; Mahmood and Ricci, 2009). The suggested information relates to various domains such as shopping items, people, documents, movies, etc. Such a system helps to bring down the complexity involved with information finding. The recommender system provides a mechanism for removing unwanted material from the retrieved information and suggests the names of experts/consultants with knowledge of the information required by the search topic or query (Ricci et al. 2011).

2.3.1. CONTENT-BASED RECOMMENDATION SYSTEMS

These systems select the items based on the similarity between the item features and the user profile. Such systems are used to suggest web results, news items, cafés, TV programs and objects for auction. However, content-based recommendations suffers from the lack of diversity problem as the recommendation in such systems is based only on the current user's preferences without including recommendations based on the preferences of other similar users. This limits the chances of exploring new items that the user might have liked but has not searched for before (Ricci et al. 2011).

2.3.2. COLLABORATIVE FILTERING (CF) RECOMMENDATION SYSTEMS

These systems are more successful and popular because they have provided solutions to the many problems of content-based filtering systems. Collaborative filtering (CF) involves the filtering or evaluation of data in accordance with the views of other people (Bell and Koren, 2007). CF technology brings together views from large web communities, and helps filter large amounts of data. Described below are the different types of CF.

2.3.2.1. MEMORY-BASED CF

Memory-based CF systems use special techniques to evaluate the similarities between users or products. The results of this evaluation are then used by e-commerce sites to recommend similar items when a particular item is purchased,

or to recommend items that have been purchased by users with similar interests. This method has been used successfully by many commercial systems (Ricci et al. 2011) owing to its effectiveness and ease of application.

The benefits of this method the following:

- It gives an explanation of the results, which is an important feature of any recommender system;
- It can be created and used easily; it is easy to add and update new forms of data;
- There is no need to study the content of the recommendation; and the tool works well with co-rated items.

This approach also has weaknesses:

- Firstly, the recommendations are created based the user rating without considering feature analysis for the recommended items.
- Secondly, as it occurs frequently with Web related products, it shows reduced performance in the case of sparse data and it cannot measure large datasets.
- Finally, it will not work for new users or new products.

2.3.2.2. MODEL-BASED CF

In these systems the recommendations are created based on models. These models are derived from user feedback using data mining and machine learning methods (e.g. regression, clustering and classification). These models can also be developed based on both expert knowledge and knowledge from research in the application domain. There are several model based CF approaches such as Bayesian Networks (Namahoot, Brückner and Panawong 2015), clustering models (Ricci et al. 2011), latent semantic models like singular value decomposition (Vozalis and Margaritis, 2007), probabilistic latent semantic analysis (Hofmann,2003), Latent Dirichlet allocation (Xie, Dong, and Hui Gao,2014) and Markov decision process based models (Durand,Laplante, Kop 2011)

The benefits of such systems include the following.

- It takes care of scattered information which was a problem for memory based CF.
- It can easily measure large sets of data.

- It gives a better forecast.
- Finally, it provides a logical basis to the given recommendations.

However, according to Lü et al. (2011) and Bell and Koren (2007), the weaknesses of this model include:

- The high cost involved in developing the accurate model.
- There needs to be a balance between its forecasting and its ability to scale the recommended information.
- Giving logical explanations of predictions is difficult for many models.

2.3.2.3. USER-BASED CF

As one of its major objectives, user-based CF recognises users with common interests. The rating given by a user for an item is used by this model to find other users with interest in the item, thus creating a pool of users. Then, recommendations are made to the users based on the ratings given by one or more users who also have an interest in that item. Thus, generally, a user-item matrix is used by a user-based CF to calculate the shared interest between users and then to make recommendations accordingly (Bobadilla et al. 2013; Bell and Koren 2007).

2.3.2.4. ITEM-BASED CF

Item based CF recognises where a user might have an interest in an item that is similar to the item required. For instance, if a user likes Canon digital cameras it is very likely that that the user likes Canon video cameras as well. Features of an item and the ratings given by other users help in getting matching products (Ricci et al. 2011; Bell and Koren 2007; Lü et al, 2011; Bobadilla et al. 2013). Benefits of item-based CF over user-based CF include the following:

- It decreases cold-start problems for new users where the users still have insufficient search or shopping history to build their profiles.
- It enhances scalability (information on similar products is more reliable than information about users who might change their interests over time).

2.4. CONCLUSION

Organisations and researchers have become more aware of the importance of enterprise information needed at different levels ranging from the management of the

day to day processing to strategic decision making. Searching for information in an enterprise involves finding the relevant people because information is not always written down and only exists in peoples' heads. However, the available enterprise search tools are inefficient and have a relatively low retrieval accuracy compared with the Internet search engines. Structural differences between the enterprise search and the web search (such as the lack of the anchor texts and the heterogeneity of the documents) make the techniques used in the web search less successful when applied to the enterprise search. The large amount of information available from the Internet as well as from the enterprise intranet, together with the poor retrieval accuracy of the enterprise search tools exacerbates the information overloading problem and limits the quality of the search result as these tools retrieve a large amount of irrelevant information.

Recommender systems sizes down the search result, omitting the irrelevant information by applying specific user-centric and/or information-centric techniques and rules. Such systems help to address the information overloading problem within the enterprise search and improve the retrieval accuracy of the enterprise search tools. The next chapter surveys the previous key research on enterprise search, relevance feedback and recommender systems, which comprises the main focus of this research and discusses a number of key intelligent used by these recommender systems.

3. CHAPTER 3 LITERATURE REVIEW

3.1. INTRODUCTION

The previous chapter provided the background for enterprise search as the problem domain. The highly competitive business environment and the large amount of information on the Internet and organisation intranets increased the importance of information as an organisational resource and created a critical need for an effective information retrieval system.

Enterprise search has a number of differences compared with web search making web search engines less efficient in the enterprise. This means that the traditional text-based search tools are more commonly used for enterprise search. However, such tools are described to be ineffective as they retrieve a large amount of irrelevant information which exacerbates the information overloading problem and limits the quality of the search result. Recommender systems have been shown to be a promising filtering tool to plug in on the top of search tools to size down the search result and in turns help to address the information overloading problem. Relevance feedback is the main data source of the intelligent recommending techniques which are used to create the required profiles and also to tune up the recommending mechanism.

This chapter discusses the previous key research on enterprise search, recommender systems and relevance feedback, which comprises the main focus of the proposed research. The main purpose of this literature review is to survey previous work on enterprise search as it is the overarching template for the proposed approach. Surveying the literature is a substantial part of any research as it helps the researcher to identify and define the research problem. It also helps the solution development and evaluation.

The rest of the chapter is organised as follows: Section 3.2 discusses previous research on the enterprise search. Section 3.3 surveys the previous work on the key intelligent recommendation approaches, including the recommender systems for enterprise search. Section 3.4 discusses the user profile including the definition and the main contents. Section 3.4 discusses the relevance feedback and how it has been used to enhance retrieval performance. Section 3.6 discusses the componential

intelligent and how these have been used to develop intelligent recommender systems. Section 3.7 presents the conclusion of the chapter.

3.2. ENTERPRISE SEARCH

The term ‘Enterprise Search’ was first coined by Hawking (2004). In his study, he introduced and defined the term Enterprise search as a different concept to the web-search. He also highlighted the main challenges to be addressed in order to achieve robustness in the search process. Since then different research has tried to address enterprise search problems in order to enhance the information retrieval performance. A database supported approach was proposed by Mangold, Schwarz and Mitschang (2006) to integrate structured information from the enterprise database and the semi-structured documents from content management systems, in order to enhance the retrieval performance. The experiments showed improvement on the recall and precision of the enterprise search.

Dmitriev et al. (2006) incorporated implicit and explicit user annotations to enhance the retrieval performance of the enterprise search by taking ideas from the PageRank algorithm which is commonly used in the web search. Implicit annotations were taken from the query logs while the explicit annotations were captured from users. The annotations were attached to the visited pages to add more relevance information to the web links. Although, the approach was shown to improve the retrieval performance slightly, it was tested on Intranet webpages which contained anchor text and did not contain heterogeneous documents, so would not be as effective on enterprise search. Recently, enterprise search has received more attention from researchers as the organisations and research communities have become more aware of its importance, and the need for intelligent approaches to address its problems (Hawking et al. 2010).

A semantic approach for the search in small and micro size enterprises was proposed by Seleng et al. (2014). They extracted hidden knowledge in emails and content management systems using tags and annotations provided by the user. The captured tags and annotations were used to build a lightweight semantic web to represent the relationships between documents required for different tasks. A knowledge cloud concept was introduced by Delic and Riley (2009). The knowledge cloud was built by extracting the keywords from enterprise information sources such as documents from content management systems, emails and database applications.

These keywords were used as candidate tags for the relevant information and filtered and ranked based on the taxonomy provided by the organisation together with Wikipedia topic headings in order to search and rank the documents. Although no results were provided the system was described to be enhancing the retrieval performance.

Bao, Kimelfeld and Li (2012) proposed an automated semantic query-rewrite rule suggestion system to help enterprise search users to write better queries. In the proposed system, the suggestions were created based on a set of rules which were extracted from the co-occurrence of the terms in the query history of successful queries. The proposed approach was shown to improve the retrieval performance and the user satisfaction as well.

In order to address the information overloading problem in the enterprise, Liu et al (2012) proposed an entity centric query expansion approach. This approach was based on expanding the user query based on the relevant entities. The entities were extracted from enterprise documents using an organisational dictionary and tags extracted from enterprise web pages and user annotations. The similarity between the user query and the extracted entity was calculated and then the relevant entities were used to expand the user query. The proposed approach was shown to improve the retrieval performance of the enterprise search.

Wang and Chen (2014) proposed a class based personalised approach for the enterprise search. In their approach, the documents were classified based on the taxonomy of the organisation and each document was assigned a particular class. During the search process the users were asked to rate the document returned by the search according to their relevance to the user query. Based on the document class and the user rating the relevance between the user and the document class was calculated. A model was then created for the user by assigning a number of classes which were used to filter the search result in the next query. The experimental results showed that the class based user model accurately represented the user interest.

Afzal and Islam (2013) presented an enterprise recommender system called Meven. The system was an Enterprise trust-based profile recommendation with privacy, which used the content from the enterprise social web to create a trust matrix between colleagues based on whether they had demonstrated similar interests and behaviour on the social web. The trust matrix then was used to bring together colleagues with similar interests and behaviour.

3.2.1. EXPERT SEARCH AND RECOMMENDATION

In medium to large size organisations, finding the relevant documents was not sufficient to satisfy the information needs of the information searcher as the information could be tacit and held only in the people heads (Suresh and Kavi Mahesh, 2006; Venkateshprasanna et al. 2011). This expanded the enterprise search task to find people who have “expert” knowledge about the query topic. However, finding such people was not straightforward and the size of the organisation, the diversity of its business and the geographical dispersion of its locations brought its own challenges and complications. As part of enterprise search, the people search inherited a number of problems which limited the retrieval performance of the search query. These problems included the heterogeneity of the document and the lack of the anchor text or internal links which were required to retrieve the required information. People search was interlinked with the document search as the document (eg. text, database records, sound tracks) was the starting point from which the people who knew most about the required topic were identified.

In the people search, people were ranked according to their knowledge of the topic or query and a list created and presented to the user. People search recently received increasing attention in the research community (Gollapalli, Mitra and Giles , 2011). It was studied by a number of researchers in different contexts including the enterprise corpora (Balog et al. 2009), sparse data university environments (Balog et al. 2007), online knowledge communities (Wang et al. 2013) and digital libraries (Gollapalli, Mitra and Giles al. 2011). People search was then categorised into profile-based and document-based approaches (Fang and Zhai 2007).

In the profile-based approach a profile was created for each user based on the documents they visited, created or authored. The user was given a rank based on matching between the profile and the given user query. Balog et al. (2009) proposed profile-based and document based approaches. The profile based approach used terms selected from the user search string to model the expertise of the users. The profiles were implemented by the vector space model and then the ad hoc model was used to retrieve and rank the users based on the relevance of their profiles to the user query. In the document based approach, a language model was employed to find the relevant people based on ranked documents. The model ranked people based on the relevance of both their profiles and the relevant documents, to the given query. The

relevance between the people and the documents was calculated based on the terms co-occurrence and the order of the co-occurred terms. The experimental results showed that the document-based model outperformed the profile-based model.

Different information retrieval models were applied in people search and recommending for the enterprise. A probability approach to rank people according to their relevance to a user query was proposed by Cao et al. (2005). The approach combined the traditional relevance model which calculated the relevance of the document to the query term based on the term frequency in the document, with the co-occurrence model which considered the co-occurrence of the query terms in the document.

The informational retrieval and graph based approaches were integrated by Deng et al. (2008) in a hybrid approach for ranking expertise. This approach integrated information from social media, online communities and forums with the document based model to rank the expertise of people for a specific topic. Combining the two approaches improved the retrieval performance beyond what was possible with each of the individual approaches. Voting techniques were borrowed from the data fusion field and applied to enhance the retrieval performance of the people search by Macdonald and Ounis (2008). The proposed voting based approach was shown to improve the retrieval performance of the people search.

Sun et al. (2013) argued that profile based methods have a lower component cost than document based methods as they used a smaller size virtual document to model the user rather than the content of the actual document. On the other hand, the document based methods were more effective in ranking people to individual documents and required less data management than the profile based methods.

PageRank (Page et al, 1999) was employed for people search. Zhou et al. (2007) used PageRank to develop a coupled walk random approach in which citation networks were combined to rank authors and documents. Wang et al. (2013) used PageRank to calculate expert authority and contribution in a specific topic in online communities. Similarly, PageRank was used to calculate the people relevance in social networks and online communities (Deng et al, 2012). The authors used comments and posts from friends' chains in social networks to estimate the importance of people in specific topic.

3.3. RECOMMENDER SYSTEMS

There has been an extensive study on recommendation systems with a myriad of publications. In this section, we aim to review a representative set of approaches that are mostly related to the proposed work undertaken in this thesis.

In general, recommendation systems can be divided into collaborative and content based recommendation. Collaborative Recommendation systems recommend an item to a user if similar users liked this item. Examples of this technique include nearest neighbor modeling by Bell and Koren (2007), Matrix Completion by Rennie and Srebro (2005), Restricted Boltzmann machine by Salakhutdinov, Mnih and Hinton (2007), Bayesian matrix factorization by Salakhutdinov and Mnih (2008), etc. Essentially, these approaches were either collaborative filtering by user or item, or a combination of these.

Collaborative filtering was used by Bell and Koren (2007) who used an algorithm to compute the similarity between users based on items they liked. Then, the scores for user-item pairs were computed by combining the scores for this item given by similar users. Item based collaborative filtering stored the information about an item liked by a particular user then recommended other items to that user if they were liked by other users (Sarwar et al. 2001).

User-item based collaborative filtering finds a common space for items and users based on a user-item matrix and combines the item and user representation to find a recommendation as shown in Fig 3.1. Rennie and Srebro (2005) and Salakhutdinov and Mnih (2008) used this approach in their research. However the user-item matrix should be factorised in order to keep its size manageable. In matrix factorisation the size of the matrix was reduced to include only those items and users which have an actual correlation. There were different approaches for user-term matrix factorisation such as factor analysis and singular value decomposition (SVD). Collaborative filtering was extended to large-scale setups by Das (2007). However it was generally unable to handle new users and new items, a problem which is often referred to as the cold-start issue.

This item has been removed due to 3rd Party Copyright.
The unabridged version of the thesis can be viewed in the
Lanchester Library Coventry University.

FIGURE 3.1: USER-ITEM SIMILARITY MATRIX (RICCI ET AL. 2011).

The second approach for recommendation systems was content-based recommendation. This approach extracted features from the item and/or user profile and then recommends items to users with preferences for those features. The underlying assumption is that users with similar preferences tend to like the same items. Linden, Smith and York (2003) proposed a method to construct a search query containing the features of items that the user liked before in order to find other relevant items to recommend.

Another example was presented by Dolan and Pedersen (2010) where the preferences of a user for particular news topics or articles were captured so that other users at the same location seeking similar topics or articles could collaborate with that user. The proposed approach used the user location to handle the cold-start problem based on the intuitive that new users should be shown the topics used most frequently in their location. This might be a good feature to recommend local news but in other domains, for example TV program recommendation, using only location information may not work as a good indication of the preferences of the user. For example, factors such as the gender and the age category might have more influence in selecting TV programs than the location. Recently, researchers have developed approaches that combine both collaborative filtering and content-based recommendations.

Melville, Mooney, and Nagarajan (2002) used item features to smooth user data before using collaborative filtering. Gunawardana and Meek (2008) used the Restricted Boltzmann Machine to learn similarity between items, and then combined

this with collaborative filtering. A Bayesian approach was developed by Wang and Blei (2011) to jointly learn the distribution of items (research papers in their case), over different components (topics) and the factorization of the rating matrix.

Handling the cold start issue in recommendation systems was studied mainly for new items (items that have no rating by any user). As previously mentioned, all content-based filtering can handle the cold start for items. Schein et al. (2002) and Gunawardana and Meek (2008) developed and evaluated some methods specifically to address this issue. Rennie and Srebro (2005) studied how to learn user preferences for new users incrementally, by recommending items that give the most information about users while minimizing the probability of recommending irrelevant content.

User modelling via rich features has recently been studied by several researchers. More precisely, it has been shown that user search queries can be used to discover the similarities between users by Song et al. (2014). Rich features from user search history have also been used for personalized web search by Song, Wang, and He (2014). For recommendation systems, Ahmed, Das and Smola (2014) leveraged the user's historical search queries to build a personalized taxonomy for recommending ads. Researchers have also discovered that the social behaviour of a user can be used to build a profile for that user. In (Abel et al. 2013), tweets from Twitter were captured for a user in order to recommend News articles to other users.

Recently, there has been an increasing interest in cross domain recommendation. There are different approaches for addressing cross domain recommendation. One approach is to assume that different domains share a similar set of users but not items, as illustrated by Sahebi and Brusilovsky (2013). In their work, the authors augmented data from the rating of movies and books together with datasets that have common users. The augmented data set was then used to perform collaborative filtering. They showed in particular that this helped the cases where users had little profile information in one of the domains (cold-start users). The second approach addressed the scenarios where the same set of items shared feedback, for example user clicks or user explicit rating, across different domains (Pan et al. 2010). In this approach the authors introduced a coordinate system transfer method for cross domain matrix factorization. Li, Yang and Xue (2009), studied the cross domain recommendation in the case where there existed no shared users or items between domains. They

developed a generative model to discover common clusters between different domains. However, a challenge in their approach was its inability to scale beyond medium datasets due to the computational cost. A different approach for author collaboration was introduced by Tang et al. (2012). In this approach they built a topic model to enable collaboration between authors working on those topics from different research fields.

For many approaches in recommendation systems the objective function was to minimize the root mean squared error on the user-item matrix reconstruction. Recently, Lee et al. (2014) developed a ranking based objective function for the recommendation of movies. The proposed function was shown to enhance the recommendation.

Deep learning techniques have been used for building recommendation systems based on collaborative filtering and content based approaches. Alakhutdinov, Mnih and Hinton (2007) used a Restricted Boltzmann Machines (RBM) model for collaborative filtering. Van, Dieleman and Schrauwen (2012) proposed Deep learning for content-based recommendation for music files. The proposed approach used deep convolutional neural networks to learn the latent features of the audio in the music files and to match these features with user preferences. The authors aimed to solve the new-item problem by substituting the lack of the user rating for the new music with latent features from the music itself.

An expert recommender system was proposed by Gao, Ilves and Głowacka (2015). The system was designed to help students to find supervisors with the right experience to help them with their final year projects. The recommendation was created based on the keywords for the academic articles which were written by the academic staff in the faculty. The system could be described as a content-based recommender system in which the reinforcement learning approach was used to model the students' interest.

Manuja and Bhattacharya (2015) used the social connection as a means to recommend items. They proposed a matrix factorization based collaborative filtering approach which uses the social connections between people to increase the accuracy of the recommendation prediction. Social connection was based on the assumption that friends have similar interests. In other words what is liked by one friend will

probably be liked by the other. The system showed a significant improvement in the prediction accuracy up to 3.09 %.

Elkahky, Song and He (2015) proposed a content-based recommendation system in which a feature set was created based on the web search history of the user . The system was mainly designed to improve the cross-domain recommendation. The experiments were carried out on recommendations for three real-world Microsoft products including Windows Apps News articles and Movie/TV programs. The results showed that the proposed approach significantly enhanced used recommendations by up to 49%. Another effort in using social connections to enhance the recommendation was introduced by Bostandjiev, O'Donovan and Höllerer (2012). They proposed an interactive hybrid recommendation approach that combined the information from social networks such as Facebook and Twitter with semantic information from Wikipedia to enhance the recommendation. The proposed system employed a hybrid mechanism that combined the collaborative filtering and content based approaches. It was also equipped with an interactive interface to explain the recommending process to the users and also to relearn the user's preferences. The experimental results showed a significant improvement on the prediction accuracy and the user satisfaction as the recommendations were explained and justified to the user.

3.4. USER PROFILE

A user profile contains a description of the user together with useful information for that user. The reason for profiling is that users are different and have different preferences and information needs which require a personalised search where the search result is customised based on the user profile.

The contents of the profile are diverse and differ based on the application domain (e.g. movies or e-shopping). In a movie recommender system. The user profile will contain the preferred actors and the genres, while in a document for an e-shopping recommender system the preferred brands will be included in the profile (Schiaffino and Amandi, 2009).

User profiles differ according to how the information was captured; either explicitly (e.g.by prompting the user to provide the required information) or implicitly (e.g. where the system learns these information by an intelligent technique) (Zu-kerman and Albrecht, 2001).

3.4.1. USER PROFILE CONTENTS

The user profile can contain a wide range of information based on the domain and the purpose of the profile. However the user profile contents could contain interests, knowledge, goals, behaviour, interaction preferences, individual characteristics and contextual information according to (Schiaffino and Amandi, 2009). These are described more fully in the following paragraphs:

- User interest is the most important component of the user profile in information retrieval and filtering systems (Brusilovsky and Millán, 2007). Interest can be a news topic, research topic, business related topic, hobby topic or a description of an item. It could be categorised as a short-term or long-term interest. For example, for the football World Cup it would be a short-term interest limited to the duration of the championship, while for research in information retrieval search engines it would be a long-term interest, because the duration is indefinite.
- Knowledge that the user has in a specific domain is an important factor. The knowledge of a domain expert could be useful for filtering information for other users in the same domain by including the expert knowledge in the filtering mechanism (Brusilovsky et al. 2005).
- User behaviour could be used to for information filtering and recommendation, where the behavioural patterns are extracted and then used to filter the information or to make recommendations. For example the user document reading behaviour (e.g. reading time and mouse movements) could be used to estimate the relevance of the document. Information could also be filtered based on the goal of the user search because the search result for a document would be different than for a product).
- Interaction preferences could be included in the user profile. These preferences could be related to the user interaction with the system interface such as the preferable message style (e.g. warning or caution), the number of the documents in the search result or the background colour (Schiaffino and Amandi, 2006). Such preferences could be used to customise the system interface to enhance the user experience. Individual characteristics such as demographics (e.g. age, gender, nationality, home address) provide useful information for information filtering and recommending ((Kolb 1984; Felder and Silverman, 1988; Honey and Mumford, 1992; Litzinger and Osif, 1993). Contextual information (e.g, current

temperature, user mood and user location) can help to sort the search results to benefit the current situation (Dey and Abwod, 1999). For example, the current location could be used in a search for restaurants, to sort the results according to the distance of each restaurant from that location.

3.4.2. GROUP PROFILES

In contrast to individual user profiles, group profiles aim at combining individual user profiles in order to model a group. Group profiles are vital in those domains where it is necessary to make recommendations to groups of users rather than to individual users.

In group profiling, the individual profiles are combined to model interests and behaviour for a group rather than for an individual user (Schiaffino and Amandi, 2009). Such a profile is useful when the recommendations are created for a group of people who share the same interest. Group profiling can be used in different domains such as tourism, movies and television programs recommendation systems. Ardissono et al. (2002) proposed a destination recommendation system, which recommends destinations to visit for tourist groups based on their shared interest, taking into consideration sub groups such as children and disabled people. McCarthy et al. (2006) suggested a collaborative approach to build group profiles. The group profiles were then used to recommend holidays to the user groups. Masthoff (2004) and Yu et al. (2006) combined individual profiles to recommend TV programmes to users based on shared characteristics such as age category and area.

In the context of the enterprise, search group profiles are useful as people in an enterprise usually work in teams or groups. These groups often need to search for information on common topics related to business processes, which make it likely that the group members could have a shared interest. Examples of groups in an enterprise could include people in the same department, project team members or people who share the same role (Lu, Huang, Jiang, 2009; Lu, Huang, Jiang, 2009; Colomo-Palacios et al. 2012).

A role based recommendation system was proposed in (Lu, Huang, Jiang, 2009). The proposed system recommended documents based on the role structure of engineering team members. Group profiles were created for design engineers, test

engineers or production engineers based on their search history. These profiles were then used to recommend documents to support the workflow of the team. Similarly (Lu, Huang, Jiang, 2010) used the employee predefined role and task along with other contextual parameters such as time, background and location to recommend information to team members. Another role based recommendation system used role based group profiles to assign project team members to project work packages (Colomo-Palacios et al. 2012). It also used a fuzzy approach to handle the uncertainty of the results when matching work package specifications with group profiles.

Due to the available data limitations the main focus of this research was to build profiles of the user based on interest and behaviour by extracting the level of interest using the user query together with implicit and explicit feedback parameters.

3.5. RELEVANCE FEEDBACK

Relevance feedback was required for personalised information filtering, particularly in recommender systems, as the user preferences need to be learned in order to build the user profiles. Relevance feedback was categorised into explicit and implicit. The two categories had some similarities as they both include, to a different extent, noise and a high level of sensitivity to search context. (Amatriain et al. 2009; Anand, Kearney, Shapcott, 2007; Hu, Koren, Volinsky 2008), but they also had a number of differences. Explicit feedback was limited while implicit feedback was plentiful. Explicit feedback was seen to be more accurate than implicit feedback in reflecting the relevancy of the retrieved object to the interest of the user. In addition, explicit feedback represented both positive and negative user judgment on the retrieved information, while implicit feedback only reflected positive judgement (Amatriain, Pujol, Oliver, 2009).

Explicit parameters were usually captured by asking the user explicitly to provide feedback to reflect the relevance of the document to their information needs. The feedback could be provided in the form of a scaled number such as “positive/inverse”, “relevant/non-relevant”, or like/dislike. Annotation and/or some form of tagging could also be used to provide more information about the viewed document (Amatriain et al. 2009).

Implicit feedback unobtrusively obtained information about the behaviour of users by recording their interactions with the system. Common techniques used to gather implicit feedback included dwell time, saving, scrolling, bookmarking, printing and

click-through, among others. Compared to explicit feedback, inferences drawn from implicit feedback were considered to be less reliable, however, large quantities of data can be gathered implicitly without incurring any additional activities by the user (Jung, Herlocker and Webster, 2007).

The relationship between user interest and relevance feedback was studied by different researchers to try and identify which implicit feedback parameters best reflected the user interest, whether there was any relationship between these parameters and also the relationship between the implicit and explicit feedback. The following paragraphs present the key researchers who have worked on relevance feedback.

Morita and Masahiro (1994) investigated user behaviour when reading news articles and how implicit relevance feedback could be used for building a profile for the user. The study was conducted on eight users who were asked to read news articles which were available on Internet discussion groups (e.g. USENET news) and then to give a rating depending on their level of interest in the articles they read. The reading time was found to be strongly correlated with the user interest, whereas, saving, following-up and copying were not found to be strongly correlated with the user interest.

Link analysis was one of the large-scaled implicit feedback approaches used in web search. Kleinberg (1999) proposed that the link of an information source for a specific topic could be called “authority” and a collection of authorities could be called a “hub”. A good hub contained a collection of good authorities and similarly a good authority was pointed to by a good hub. This principle was then applied successfully in the PageRank algorithm applied by Google.

Claypool and Mark (2001) suggested a categorization for relevance feedback parameters which included both implicit and explicit parameters. In their research, they tried to address the essential question of what implicit parameters could be used as indicators of user interest. The study was carried out on 75 students where the students were asked to use a customised web browser for unstructured browsing. The browser was designed to capture certain implicit parameters (e.g mouse clicks, combined scrolling, and time on page). The browser also captured the explicit rate of relevance for each visited page. The explicit rate was used to evaluate implicit parameters as interest indicators. Both the time spent on a page and the amount of scrolling on a page were found to be strong indicators of interest as they had a

positive correlation with implicit ratings. However the mouse clicks and individual scoring indicators were found to be ineffective as a means of predicting the explicit rating.

Oard and Kim (2001) suggested that the user models could be developed automatically based on the implicit relevance feedback. They classified the implicit feedback parameters into examine, retain, reference, and annotate categories. And these main categories could then be further subdivided based on the scope of the visited information (e.g. segment, object, or class). The study was carried out using academic and professional journal articles and abstracts. It was found that the printing and reading time could be used as strong evidence of the relevance level of the article and also that the user spent a longer time reading academic articles than news stories. This classification framework is shown in Fig 3.2.

This item has been removed due to 3rd Party Copyright. The unabridged version of the thesis can be viewed in the Lanchester Library Coventry University.

FIGURE 3.2 : OARD AND KIM (2001) CLASSIFICATION FOR POTENTLY OBSERVABLE BEHAVIOUR (IMPLICIT FEEDBACK PARAMETERS)

Reading time was also examined as a document re-ranking technique by White, Ruthven and Jose (2002). The proposed technique used the reading time captured from the user's interaction with the search results to automatically re-rank the retrieved documents and then update the display to reflect the ranking of the documents. The retrieved documents were displayed to the user as a summary of the sentences in the document. Based on the assumption that the users spend a longer time on the relevant

document summaries the search result was updated based on the captured reading time. This assumption was investigated and it was found that users spend significantly longer time viewing relevant academic article summaries than the news stories.

Kelly and Teevan (2003) added a new behaviour category to the Oard and Kim's framework and called it "Create". The category includes the implicit parameters relating to the user behaviour when creating a new information item or updating an existing one. They also added some additional parameters to the existing categories which as shown in Fig 3.2.

This item has been removed due to 3rd Party Copyright. The unabridged version of the thesis can be viewed in the Lanchester Library Coventry University.

FIGURE 3.3: KELLY AND TEEVAN (2003) EXTENDED CLASSIFICATION FOR IMPLICIT FEEDBACK PARAMETERS

It has been argued that click-through data does contain useful information indicating likely relevance as users generally do not randomly click on links. Fox et al. (2005) measured user activity and collected explicit relevance judgements based on Web search. They found the best retrieval model was the combination of click-through, dwell time and the way a user ended a search session.

Click-through data in the web search was examined as a reliable source of implicit feedback by Ramachandran (2005) the clicks were found expressive, but biased, which affected their ability as an absolute indicator of user interest. However, the relative user preferences which were derived from the clicks were found to be relatively accurate. This notion is supported by other studies that showed the positive effects of

click-through data in estimating the interests of users (Bidoki et al 2010; Jung, Herlocker and Webster. 2007; Smyth et al. 2005).

Poblete and Baeza-Yates (2008) proposed an approach to incorporate implicit feedback information into the vector space representation of the retrieved document. Rather than representing a document based on TF-IDF (term frequency – inverse document frequency) which is stripped of user information need; a document could be represented using the vocabulary of the user during the retrieval of a particular document. If a user entered a search query and clicked on the document then the query was going to represent the document even if some vocabulary from the query was missing from the document. Based on such representation a similarity analysis of the documents can be performed.

Page review literature refers to ‘‘re-finding’’ as post-click behaviour in which users return to the same web pages that have been visited. Tyler and Teevan, (2010) studied this behaviour as an indication of user interest and they found that about 38% of all user queries were used to re-find a previously visited page. The experimental results showed that queries which were created to re-find a page were better than the ones that were previously created to find the page. Similarly, Tyler, Wang, and Zhang (2010) proposed the re-ranking method which ranked higher the re-finding URLs. The experiments showed that the retrieval performance could be enhanced using re-finding based predictions for the relevant page in the personalised search.

Post-Click Behaviour (PCB) term was introduced by Guo and Agichtein (2012) to indicate the behaviour of users during the dwell time (time spent reading the information retrieved). The experiments showed that post-click parameters, such as mouse movement on the page and combined scrolling, together with the dwell time are useful for enhancing document relevance prediction. They proposed a method for identifying the patterns of examination and interaction behaviour associated with viewing relevant or non-relevant documents. The extracted patterns were then used to estimate the relevance of the visited document. The proposed method was shown to be more effective in estimating the document relevancy than using dwell time alone.

White and Buscher (2012) proposed that text selection actions on the visited page could reflect the user interest level in the page and so could be used to enhance the retrieval performance. The proposed approach used the text selection events performed by the user on the page as an indication of the level of interest and then used the associated text such (e.g. the title, the snippet, or a combination of both of these), with

those events to model the user interest. This approach was shown to improve the retrieval significantly.

Buscher et al. (2012) analysed user behaviours such as clicks, hovers, text selection and cursor trails on the search engine result page (SERPs), and used this information to cluster the users based on the similarity of their behaviour. Balakrishnan and Zhang (2014) proposed an integrated implicit feedback model to improve the post-retrieval document relevancy. They combined dwell time, click-through, page review and text selection. Their study found that using all these parameters in a single model provides advantages over just using dwell time, click-through, page review and text selection alone. Additional statistical tests that were conducted revealed the differences to be significant. Furthermore, it was also found that text selection had the highest accuracy compared to other techniques, including dwell time and click-through, the two common techniques that have been extensively researched. This indicates that user's post-click behaviours can be efficiently used to improve document relevance.

The examination of the literature on relevance feedback shows that there are several implicit parameters that can be used as indicators of the interest level of the user in assessing the document relevance. However, there is no consensus on a specific combination of parameters to be used to estimate the user interest or relevance levels for a document or an item. User behaviour could change following changes to the search environment or the nature of the required information. Also, there are behavioural differences between individuals which could lead to differences in interest levels

3.6. MACHINE LEARNING FOR RECOMMENDER SYSTEM

In order to develop inelegant recommender systems, machine learning methods were used to train the computer program on example data in order to solve a given problem (Alpaydin 2014). Alpaydin (2014) further describes classification as one of the main tasks in machine learning where the computer program classifies the data instances into classes based on learnt classification rules (Alpaydin 2014). There are a number of classifying methods which have been proposed by the research community and used in developing recommender systems such as decision trees, Bayesian methods, artificial neural networks and fuzzy rule based classification.

Decision trees are rule based classification methods in which the rule base is viewed as a tree-like graph where the conditions are the parent nodes and the class are

the leaves (Witten and Frank 2005). The result of each parent node is a logical value of true or false which determines the path from the root node to the leaf. This method could be described as simple, easy to implement and mainly used for comparative purposes. However, it performs more effectively in the cases where the boundaries between the data categories are clear and do not include a high level of uncertainty.

There are a number of decision tree based recommender systems that have been proposed. A collaborative filtering recommender system was proposed based on a dedicated decision tree. More precisely, for each new item in the database a prediction for the customer feedback (Like/Dislike) on the features of the item was created based on the previous feedback from the customer and other customers who had previously expressed a preference for that feature (Breese et al. 1998). Decision trees were also used to develop content-based recommender systems in which each customer was represented as a decision tree that contains the features of the preferred items. These features were then used to model the user preferences and splitting criteria (Li and Yamda 2004, Bouza et al. 2008). In another research, a hybrid approach was proposed based on decision trees, the approach combined the item tree and user tree into one decision tree which was based on the user preferences for particular item features in addition to collaborative feedback from similar users (Gershman et al 2010).

Naïve Bayes classifiers are simple probabilistic classifiers which are based on Bayes' theorem with a naïve independent assumption (Alpaydin 2014). They assume that the values of features are independent of each other and each feature is considered to contribute independently to the probability of the class. In other words, the classifier ignores any correlation between the features in identifying the class. Naïve Bayes classifiers have been used in different recommender systems (Melville, Mooney and R. Nagarajan 2002; Ghazanfar and Prugel-Bennett 2010; Namahoot, Brückner and Panawong 2015). Melville, Mooney and R. Nagarajan (2002) used a Naive Bayes classifier to build the profiles of users and movies into a hybrid recommender system to recommend movies to users. The Naive Bayes classifier was used to approximate the level of interest between the users and the movies in a user-item matrix. Ghazanfar and Prugel-Bennett (2010) combined the Naive Bayes classifier with collaborative filtering to achieve better accuracy in predicting the movies for which the user showed an interest. The Naive Bayes classifier was used to identify the similarities between the users profiles based on

similarity and closeness of the features. In a recent research, Namahoot, Brückner and Panawong (2015) used a Naive Bayes classifier to develop the Tourism Recommender System (CAT-TOURS) that support the tourist in making a decision. The system combined the Naive Bayes classifier with a tourism ontology to increase the accuracy of the classifier

The Naive Bayes classifier, in general, is shown to be highly scalable as it is fast to train and to classify based on the given data. However the assumption of feature independences affects the interpretability of the classifying rules (Cintra et al. 2009; Kantor et al. 2011; Alpaydin 2014). Also the approach is not shown to perform well on uncertain data (Cintra et al. 2009; Kantor et al. 2011; Alpaydin 2014).

An artificial neural network, is a computational model that intends to simulate the structure and/or functional aspects of biological neural networks (Cintra et al. 2009; Kantor et al. 2011; Alpaydin 2014).

In the artificial neural networks, the classifying function is constructed in way which simulates biological neural networks (Cintra et al. 2005; Kantor et al. 2011; Alpaydin 2014). This technique was used by Lee and Woo (2002) in their collaborative filtering recommender system to classify the interest level of the user in the features of a movie. The users in the proposed system were grouped based on their demographical characteristics and within each group the users were clustered based on their preferences. These preferences, together with features of the movies were then used by the artificial neural network classifier to identify whether or not the movie should be recommended to the user.

Similarly, Christakou and Stafylopatis (2005) used trained artificial neural networks in a hybrid movie recommender system to build a user-item matrix to classify movies based on the user preferences and the features of the movies. Ren et al. (2008) used an artificial neural network classifier to approximate the relevance for a user of the features of items for shopping applications. The relevant features were used to create the user profile which in turn was used to recommend the items.

Artificial neural networks are usually used for data modelling in the cases where the relationship between the inputs and outputs are complex. They also can be used to find the common patterns with which to classify the data. Artificial neural network classifiers are said to have a high tolerance for the noise in the data and are able to handle the uncertainty in order to classify data on which they have not been trained. However, interpretability is one of the main problems of this approach and the neural

network based approaches are considered black box (Zhou 2004; Cintra et al. 2009; Kantor et al. 2011).

Fuzzy classifying methods were used in machine learning task such as DoC-Based Method (Michalewicz 1996) and Wang & Mendel Method (Wang 2003). The Fuzzy method usually uses the fuzzy sets to convert the numerical values of the features and classes to linguistic labels such as “High”, ”Medium” and “Low” based on the value of the membership function. This label is then used to determine the extent to which that numerical value belongs to the fuzzy set. The combination of the linguistic labelled features and classes form an **If->Then** rule base that could be compressed in a group of common patterns which are used to classify the data.

Fuzzy logic systems (FLSs) have been applied to a range of application areas in Information Retrieval (IR) that include personalised search and Recommender Systems (RS). A single individual fuzzy based recommending method is proposed (Yager, 2003) called the “recursive method” in which the recommendation is created recursively and based on the profile of the user without making use of any other collaborative preferences. The fuzzy sets were used to model the recommended object as well as justifying the recommendations. Cornelisa et al. (2007) proposed a fuzzy based conceptual framework for recommending one-and-only items. One-and-only items are the items which have only one occurrence in the data where the single occurrence of such items limits the ability of the classic collaborative filtering abilities to recommend the required item. Fuzzy logic was used for modelling user preference to justify the similarity calculation.

Carbo and Molina (2004) developed a collaborative filtering based algorithm in which the linguistic labels and the associated fuzzy sets were used to handle the uncertainty and inaccuracy in ranking and recommending items. A hybrid fuzzy based approach combining content-based and collaborative filtering was developed to recommend movies to individuals based on the details of the movies, and the preferences of other users, (Perny and Zucker, 2001). Here, fuzzy logic was used to model the similarity of the preference of individuals for a specific movie). Later, a similar approach was proposed by Campos, Fernández-Luna and Huete (2008).

Doctor, Roberts and Callaghan (2009) proposed a fuzzy based agent to rank and to recommend the CVs for suitable candidates within a recruitment system. Fuzzy logic

was used to model the preferences of the selection board members for the required job and also to resolve the uncertainty and conflict in the group decision making.

A fuzzy based method that improves the collaborative filtering efficiency in the context of multiple collaborating users was proposed by Eckhardt (2012). In this system the user was modelled using fuzzy sets in order to handle the bias and the uncertainty which may result when involving different users.

Recommender systems, in general, rely heavily on the relevance feedback as a main data source to create the profiles as well as to build and tune up the recommending mechanism. This requires the classifier to have the capability to handle the uncertainty that the relevance feedback include in order to increase the accuracy. It also should provide a transparent and interpretable classifying method by which the decision could be justified in order to model the user judgment and decision making process. User interest might change from time to time which means the classifying mechanism should be adaptive and responsive to these changes in order to reflect the actual current user interest behavior.

Based on the critical review of a number of approaches, the fuzzy based approach was found to be capable in dealing with uncertainty in the data set and in the meantime provides interpretable data model. Therefore, this approach will be further investigated and used as part of the proposed approach Fuzzy logic enables the modeling of uncertainty in the data which is one of the common features of user feedback by molding the proximity between a data instance and particular class. Furthermore, fuzzy based approaches produce an interpretable fuzzy rule base by which the result could be justified and reasoned and even compared with the common sense expert knowledge. There are some other methods that can model the uncertainty, such as the artificial neural network methods, but as discussed above these method do not produce interpretable classifying rules which justify the result. On the other hand there are some other methods which can produce interpretable rules such as decision tree methods, however, they are not designed to handle the uncertainty and complexity associated with the data. Naïve Bayes methods are described as highly scalable and fairly accurate but they do not create an interpretable result or consider any correlation between the features of the data.

In addition fuzzy logic is linked to the rule induction which is a machine learning area in which a rule base is extracted from a set of observations (Ishibuchi and Yamamoto, 2005; Cintra ,etal., 2006); Wu, Mendel and Joo, 2010) The extracted rule base provides a semantic representation of the relationships in the data and also represents the common patterns. Fuzzy rules could be extracted from the observations by applying a rule induction techniques such as association rule learning. This creates an adaptive fuzzy classier in which the fuzzy rules change based on the changes in the data instances (Ishibuchi and Yamamoto 2005; Cintra et al. 2006; Wu, Mendel and Joo 2010).

However, the aforementioned fuzzy approaches for recommender systems were mainly focused on the identification of indicators of document relevancy and user preferences. They did not consider combining these indicators with profiles of the user and/or task which may have specific relevance to the information needs of the enterprise. Also, many of these approaches focussed on well described content such as news, stories, events, and movies and not on the unstructured content (documents) commonly found in enterprise systems which contain less descriptive detail. In addition, such systems also had to contend with uncertainties (subjectivity and inconsistencies) in information relevance and the user's perception of relevance in relation to the retrieved content. Finally, these approaches did not integrate implicit and explicit feedback parameters with the query text analysis in order to gain more reliable relevance feedback.

3.6.1. FUZZY LOGIC

Fuzzy Logic, a multi-valued logic, was developed in 1965 by Lotfi A. Zadeh (Zadeh, 1965; Zadeh, 1973; Mamdani, 1974). It enabled intermediate values to be explained in the midst of traditional evaluations like true/false, yes/no, high/low, etc. Ideas like “somewhat longer” or “very quick” can be expressed statistically and administered by PCs, in order to provide a human approach in the computers processes (Zadeh, 1984).

3.6.1.1. FUZZY SETS AND CRISP SETS

Fuzzy sets are a fundamental concept in fuzzy logic. They are used to model the cases where the categories of the data had no clear or fixed boundaries. Fuzzy sets

are different to the conventional Bivalent crisp sets (Zadeh, 1965; Zadeh, 1973; Mamdani, 1974). In the crisp set the data range was decomposed into fixed and not overlapped sub ranges and each sub range was given a label. For example, the possible values of the relevance weight r of a term to atopic are the set H of the real numbers between 0 and 1. A sub set of L called A can be defined for the r values between 0 and 0.2. then the characteristic function of the set L is assigned a value of 0 or 1 to each number of the set X based on whether they are members of L or not as is shown in Fig 3.4.

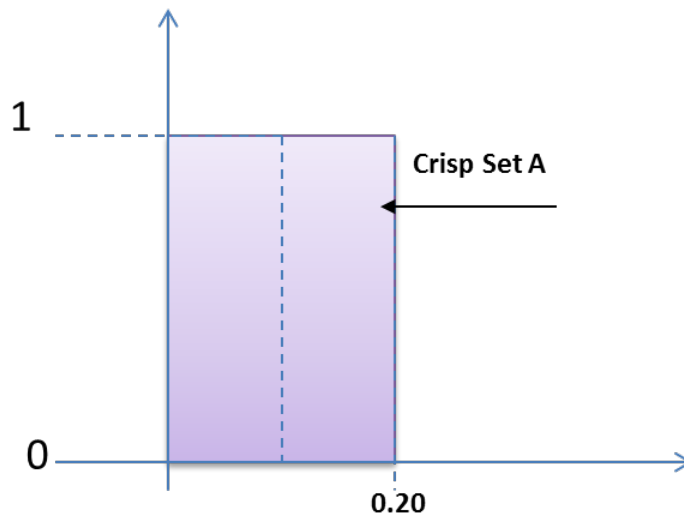


FIGURE 3.4: CRISP SET

Assuming that the set L reflects a low relevance between a query term and a specific topic, then only the terms which have r value between 0 and 0.2 will be given the value 1 by the characteristic function and categorised as low relevance. This method in categorising data might be adequate in some cases where the boundaries between the sub sets are clear and fixed such as the university grading system which identifies the student mark as pass, merit or distinction. However in some other cases Crisp sets are not adequate to identify the participation of a data instance one of the sub sets, especially when the boundaries between sub sets are loose and overlapped. An example of these cases is the same relevance weight from the last example; let's assume that different people were asked to provide the relevance weight (r $0 \leq r \leq 0.1$) and the whole range was divided in the following subsets : $VL(r$ $0 \leq r \leq 0.1$) to express very low, $L(r$ $0 \leq r \leq 0.1$) Low , $M(r$ $0 \leq r \leq$

0.1) to express Medium , $H(r \ 0 \leq r \leq 0.1)$ to express High and $VL(r \ 0 \leq r \leq 0.1)$ to express Very High. In this case it should classify the value 0.21 in one of the two sets, Very Low and Low as the value is just on the boundary between them. .This uncertainty is because the value 0.21 might mean very low for one person and low for another, based on their subjective judgment. As shown in Fig 3.5, Fuzzy sets can model these cases as they allow subsets to overlap which enables the data instance to participate in two different sets with different memberships.

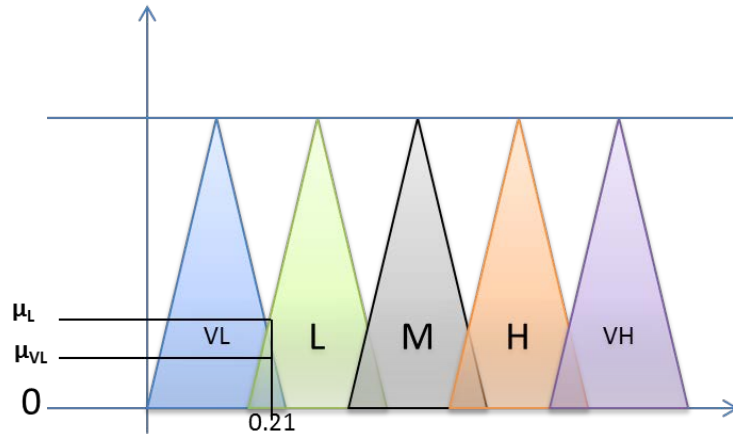


FIGURE 3.5: FUZZY SETS VL, L, M, H AND VH

Membership is a numerical value between 0 and 1 which indicates to what degree a particular data instance participates in a set. After identifying the data instance it is categorised by fuzzy operations. Fig 3.6 shows the difference between Crisp and Fuzzy sets.

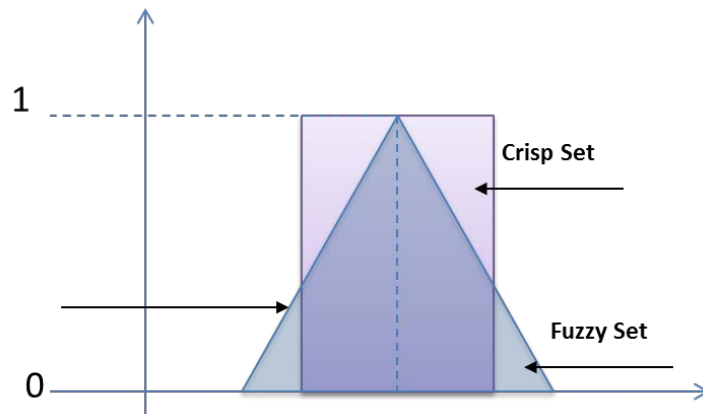


FIGURE 3.6: FUZZY SET AND CRISP SET

Fuzzy sets extends the $[0,1]$ notion of the characteristic function in the Crisp sets by permitting additional values among 0 and 1 based on the shape of the

membership function. As shown in Fig 3.7, Member functions for fuzzy sets can be defined in different ways such as Trilateral, Trapezoid, S-Curve, Singleton and others. The Shape of the membership function used defines the fuzzy set and interprets the membership value.

This item has been removed due to 3rd Party Copyright. The unabridged version of the thesis can be viewed in the Lanchester Library Coventry University.

FIGURE 3.7: MEMBER FUNCTION SHAPES B (UNIVERSITY OF STRATHCLYDE 2015)

Formally, the fuzzy set could be defined as:

“Let X be a space of points, with a generic element of X denoted by x . Thus $X = \{x\}$.

A fuzzy set A in X is characterized by a membership function $f_A(x)$ which associates with each point in X a real number in the interval $[0,1]$, with the values of $f_A(x)$ at x representing the "grade of membership" of x in A . Thus, the nearer the value of $f_A(x)$ to unity, the higher the grade of membership of x in A “ (Zadeh, 1965).

3.6.1.2. OPERATIONS ON FUZZY SETS

In fuzzy logic the crisp sets logic operations x OR y , x AND y and NOT OT are replaced with $\text{MAX}(\mu_A, \mu_B)$, $\text{MIN}(\mu_A, \mu_B)$ and $1 - \mu_A$. where μ_A is the membership function of fuzzy set A and μ_B is the membership function of fuzzy set B . Fig 3.8 shows the fuzzy sets, A and B . and Fig 3.9 shows the operations OR and AND on those fuzzy sets.

This item has been removed due to 3rd Party Copyright. The unabridged version of the thesis can be viewed in the Lanchester Library Coventry University.

FIGURE 3.8: FUZZY SETS A AND B (UNIVERSITY OF STRATHCLYDE 2015)

This item has been removed due to 3rd Party Copyright. The unabridged version of the thesis can be viewed in the Lanchester Library Coventry University.

FIGURE 3.9: FUZZY OPERATION EXAMPLE (UNIVERSITY OF STRATHCLYDE 2015)

3.6.1.3. FUZZY CLASSIFICATION

Fuzzy classifiers are based on a set of rules which reflect the association patterns between a set of inputs or features (antecedents) and outputs or classes (consequents). Fuzzy rules take the form of *IF* \rightarrow *THEN* rule for example “*IF feature A low AND feature B medium AND feature C medium AND feature D medium THEN Class = class 4*”.

TABLE 3.1 : FUZZY RULES EXAMPLE

R#	feature A	feature B	feature C	feature D	class
1:	Low	Medium	Medium	Medium	Class1
2:	Medium	High	Medium	Low	Class2
3:	Low	High	Medium	High	Class3
4:	Low	High	Medium	High	Class 1
5:	Medium	Medium	Medium	Medium	Class 4
...:
N:	Low	High	Medium	Low	unknown

This section provided discussed the fussy logic briefly. However more details on the construction of a fuzzy system will be discussed in **Chapter 5** while explaining the proposed approach.

3.7. CONCLUSION

The importance and large amount of information relating to enterprises within organisations and available on the Internet have together created a critical need for an effective approach to retrieve the required information as quickly as possible. Enterprise search has a number of differences compared to the web search, which

makes web search engines less efficient when used in the enterprise. This means the traditional text-based search tools are more likely to be used for enterprise search.

However, such tools are described as ineffective because they retrieve a large amount of irrelevant information which exacerbates the information overloading problem and limits the quality of the search result. Although enterprise search has recently received more interest in the research community, the number of research studies are relatively limited and this area requires more studies to be performed to identify the most appropriate approach.

Recommender systems performed well in different fields in the web search such as movies, e-shopping and books which make a promising filtering tool to plug in on the top of a search tool to size down the search result and in turn helps to address the information overloading problem.

Relevance feedback is the main data source of the intelligent recommending techniques which are used to create the required profiles and also to tune up the recommending mechanism. The user in the enterprise search is described as having more of a tendency to provide feedback than the web search user because the enterprise user is using the search for a particular organisational task and requires relevant information. Also the privacy and security concerns are considered to be less significant in the enterprise search because the search tools are provided by the organisation and so theoretically should be safe and secure. This encourages the use of relevance feedback for information filtering in order to enhance the retrieval performance of the enterprise search.

The next chapter discusses the user study and experimental design. The user study was the main source of relevance feedback parameters which were used to develop the proposed approach as well as to test the performance of the model.

4. CHAPTER 4 USER STUDY

4.1. INTRODUCTION

The literature review in chapter 3 shows that relevance feedback has been used as a useful tool to enhance the search experience. It is a substantial data source for search personalisation and recommendation systems that are widely used to tackle the information overloading problem. There also exist several relevance feedback parameters that can be used, both as indicators of the interest level of the user in the documents or the document relevancy. However, there is no consensus on a specific combination of parameters to be used to estimate the user interest or the relevance levels of a document or item. This chapter discusses the user study which was conducted as a part of the research to capture the relevance feedback from 35 users, during their searching process, based on an enterprise document test collection. The relevance feedback was captured in order to maintain an adequate amount data for the profiling process in the proposed approach. It also provided the means to conduct empirical research, to gain a better understanding of the nature of the relationship between the implicit feedback, and determine the relevance level of the document within the context of the enterprise.

The rest of the chapter is organised as follows: Section 4.2 introduces the user study and discusses the participants, dataset and search tasks. Section 4.3 discusses the user study experimental setups including the search system components and indexing process. Section 4.4 discusses the data collection and the captured data. Section 4.5 is the conclusion of the chapter.

4.2. USER STUDY

The study was carried out using a controlled observation technique (Magnusson et al. 2009; Gulliksen et al. 2003) in which 35 users were selected randomly and invited to complete a group of 20 simulated search tasks which were designed to simulate real world search scenarios in the enterprise. The participants were informed about the objective of this user study, briefed about the procedure and given protocols of these

tasks for the selected users to complete. During the search process the system automatically captured the implicit and explicit parameters in addition to user queries.

4.2.1. PARTICIPANTS

A group of 35 users were invited to participate in the experiments. Each user was briefed about the research aim and objectives, as well as the purpose of the user study. The experiment procedure was explained to them, including the steps within the experiment, the estimated time to complete the steps and how the captured data would be used in the research. The search tasks and the corresponding information sheet were given to the users to read. The tasks were then explained to them which included an introduction about the company CISOR, the role of the Science Communicator and the role responsibilities. The users were then trained to use the provided system and asked to freely formulate their queries to search for information to help them find solutions to those tasks

The 35 users were unpaid participants, consisting of 10 females and 25 males. The users were of two occupations: university students and university staff members, and they were of different disciplines and qualified with different degrees (including undergraduate postgraduate and doctorate). Table 4.1 shows the participants characteristics.

4.2.2. DATASET (TREC Enterprise Trak 2007)

The user study was carried out using the TREC 2007 Enterprise Track dataset. This dataset was selected because it is one of the most common test collections for enterprise search. It consists of real world organisation (CSIRO) data. CSIRO, the Commonwealth Scientific and Industrial Research Organisation, is the Australia's national science agency and one of the largest and most diverse research agencies in the world. It is an organisation distributed over 50 sites in Australia and around the world. It has 17 divisions, and has conducted research in areas such as entomology, industrial physics, mining, sustainable ecosystems and information & communication technologies.

TABLE 4.1: PARTICIPANTS' CHARACTERISTICS.

<i>Characteristic</i>		<i>#Of Participants</i>
<i>Gender</i>	<i>Female</i>	10
	<i>Male</i>	15
<i>Age</i>	<i>(20-30)</i>	24
	<i>(30-40)</i>	11
	<i>(40-50)</i>	1
<i>Occupation</i>	<i>University Student</i>	24
	<i>University Staff Member</i>	11
<i>Education</i>	<i>Secondary School</i>	6
	<i>Degree</i>	8
	<i>Masters</i>	14
	<i>PhD</i>	2
<i>Nationality</i>	<i>UK</i>	5
	<i>Jordan</i>	20
	<i>Iraq</i>	4
	<i>Malaysia</i>	3
	<i>Romania</i>	1
	<i>US</i>	2

The test collection is designed to reflect the enterprise search environment where the documents are heterogeneous (Bailey et.al, 2007). The dataset consists of 370,715 documents, with a total size 4.2 gigabytes. The corpus contains different types of documents such as html, text, pdf and others. The data set is labelled as the test collection and provides a group of 50 queries which were previously created by real users and associated with the relevant documents for each query according to the judgement of users (Bailey et.al, 2007). *Fig 4.1* shows a sample of the provided labelled data

```

<top>
<num>CE-001</num>
<query>genetic modification</query>
<narr>
Over arching information on gene technology / biotechnology. Specific pages on certain GM (e.g.
cotton).

```

```

</narr>
<page>CSIRO135-03599247</page>
<page>CSIRO141-08973435</page>
<page>CSIRO141-07897607</page>
</top>
-----
<top>
<num>CE-002</num>
<query>hairpin RNAi / gene silencing</query>
<narr>
Information to help scientists find out more about hairpin RNAi technology. Specific contacts to obtain
vectors.
</narr>
<page>CSIRO197-05231046</page>
<page>CSIRO139-13111797</page>
<page>CSIRO145-13752815</page>
</top>

```

FIGURE 4.1: SAMPLE OF THE LABELLED DATA

4.2.3. SEARCH TASKS

The labelled data set consists of only 50 queries and provides a very limited relevance feedback which is the relevant documents for each query. It does not provide any implicit or explicit relevance feedback. Hence, there was a need to extend the labelled data to include more user queries and relevance feedback. The labelled dataset was extended by creating 20 simulated search tasks to be given to the users to complete and the provided system search system collected the relevance feedback during the search process.

The search tasks were designed to simulate real world search tasks within the organisation. Departmental tasks for example would be taken from enterprise systems (e.g. Human Resources application) whereas Project based tasks would be taken from project management systems (e.g. work breakdown application). The purpose of linking user search to the required task is to build a task based profile which reflects the global interest of a group of users rather than an individual user and in turn helps to make recommendations based on collaborative filtering as will be discussed in chapter 5.

In the designed tasks, the user is acting as a science communicator working for CSIRO. The Science communicator is one of the important roles in CSIRO which is to enhance CSIRO's public image and promote the capabilities of CSIRO by managing information and interacting with industry groups, government agencies, professional groups, media and the general public. The description of this role shows the need for people who are working as communicators to obtain sufficient information in the right time. As a communicator, the user was asked to provide a number of information retrieval tasks for specific topics related to the organisation's business processes. The search tasks were developed based on 25 of the provided queries and their corresponding narratives. The narratives provided useful information on the context of the query and the information needs for which the query was created. The motivation behind using the provided queries to develop the tasks was that the relevant documents for the used queries could be used for evaluating the proposed approach in later stages of the research. The relevant documents in the dataset are required to evaluate the search systems' performance.

Fig 4.2 shows an example of a provided query, which corresponds to the narrative and relevant documents, together with the developed task based on this query.

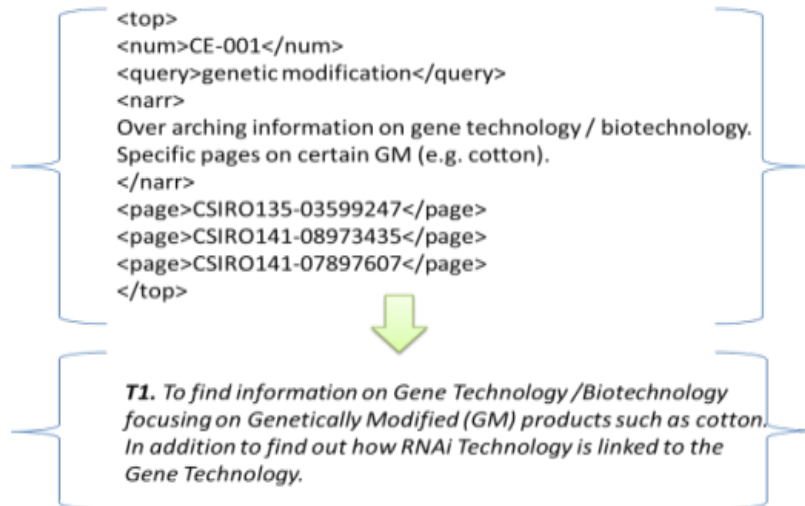


FIGURE 4.2: SAMPLE OF THE DEVELOPED TASKS

The estimation of the task duration was achieved by giving each task to three different people to finish and then timing the duration of the task for each of them. Then, by taking the time average of the three the people for each task, the result showed the estimated time averages for the tasks to be from 5 to 10 minutes.

4.3. USER STUDY EXPERIMENTAL SETUP

An enterprise standard search system was implemented and deployed to run the user study. However, the system was implemented only for the purpose of carrying out the user study and not as a part of the proposed approach. The enterprise standard search, which could be any enterprise search engine (e.g. Solr, Oracle or Microsoft) acts as a front-end to the proposed approach in order to extend the capabilities of the search for the enterprise user. The system was developed to allow users to search for the provided tasks and also to capture the relevance feedback (both implicit and explicit) from the users. In order to make the system easy to access for the participants the system was deployed on a web server and hosted by an Amazon Web Cloud service. Fig 4.3 shows the overarching architecture of the system.

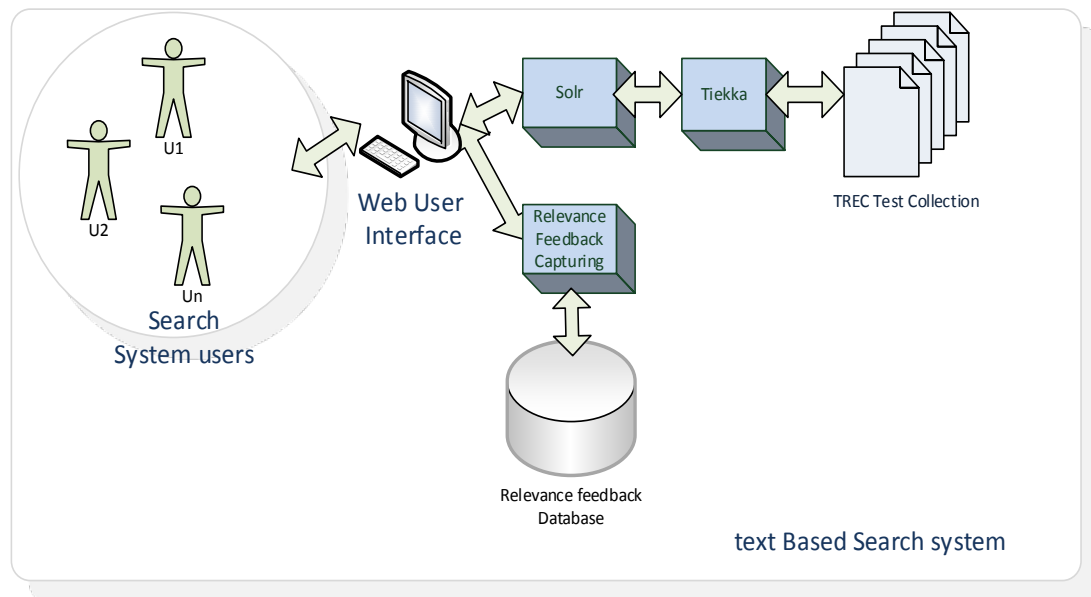


FIGURE 4.3: USER STUDY EXPERIMENTAL SET UP

4.3.1.1. WEB USER INTERFACE

A web-based graphical user interface was developed to support the purposes of the study. It was implemented in Java and was compatible with well-known web browsers such as Internet Explorer, Google Chrome and Firefox. The interface allows the users to sign on using a login screen:

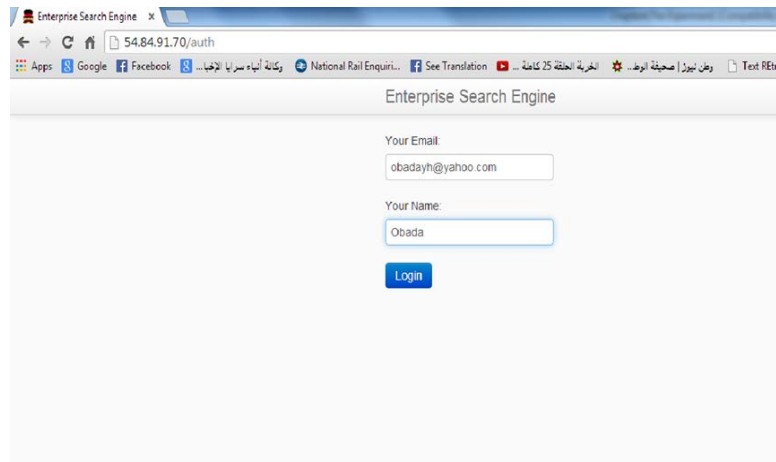


FIGURE 4.4: LOGIN SCREEN

The user needs to provide a unique email address and name (e.g. an alias name). The provided email address is then used by the system to identify the user.

After the user has logged on, they will choose the required search task.

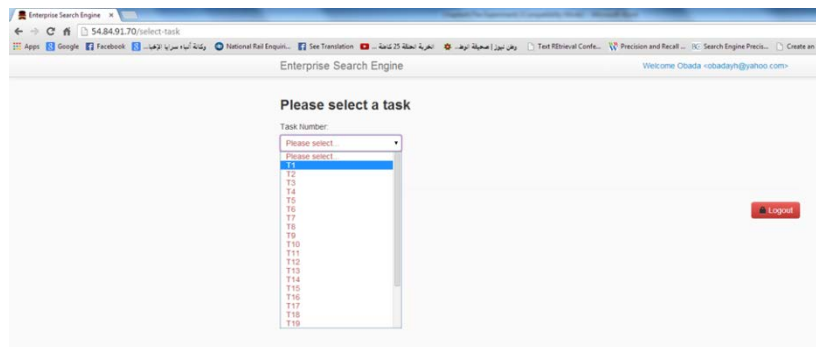


FIGURE 4.5: SEARCH TASKS SCREEN

After selecting the required task, the search screen will appear to allow the user to apply their own queries in order to complete the task. As shown in *Fig 4.6* the user has the ability to create the query and then press the button search and the result will then appear as a list of documents. The list could then be sorted by relevance, date or alphabetical order by title. The list could also be filtered based on the type of the document (e.g. pdf, doc, ppt). Finally the user can finish the task by clicking the button 'Finish Task' on the top right corner. The user can redo the task if necessary.

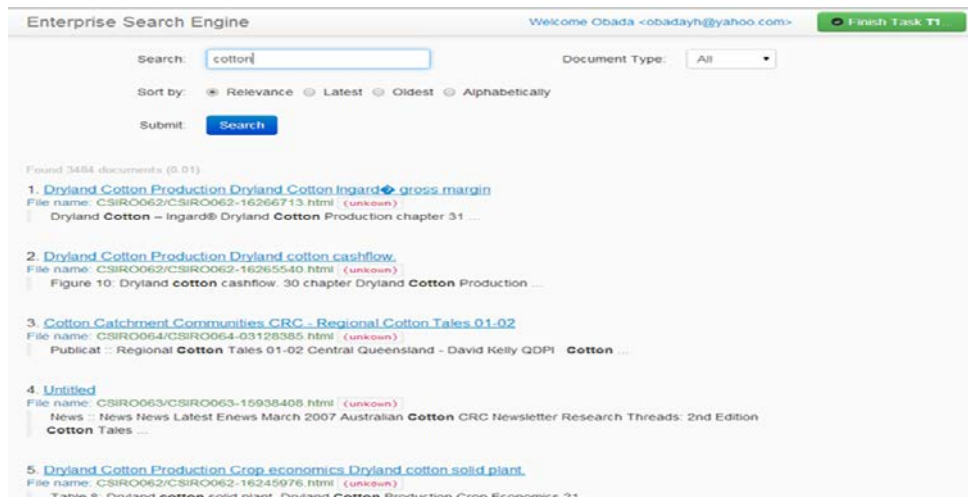


FIGURE 4.6: SEARCH SCREEN

Once the user selects any of the documents in the search result, the document screen appears. As shown in Fig 4.7 the user has the ability to download, bookmark and print the document. The user have the choice to either; enter feedback on how relevant the document is to their query by selecting the button 'Feedback' or; choose to redirect back to the feedback screen once they leave the page.

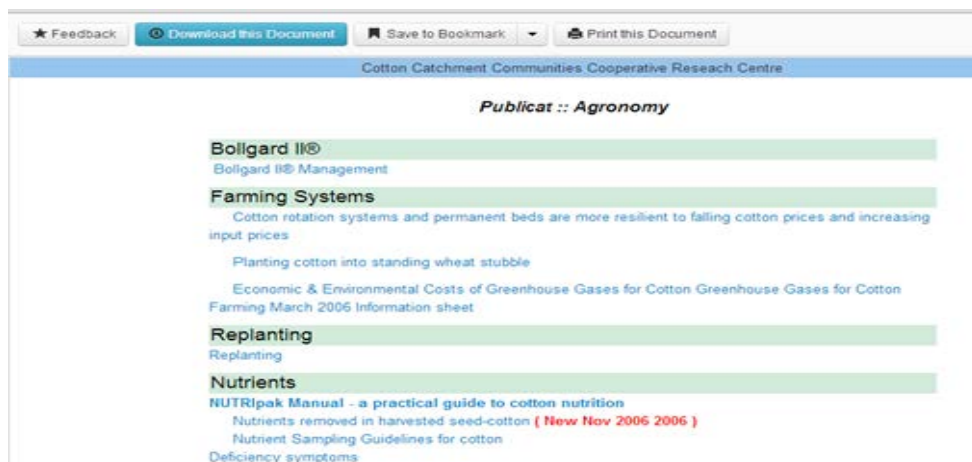


FIGURE 4.7 : DOCUMENT SCREEN

After the user has finished reading the document, they are asked to provide their judgment on the document relevancy. As shown in Fig 4.8 the user is prompted to select a number from the list to indicate how relevant the document is to their query. The list consists of integer number between 1 and 10. After submitting the feedback the system returns the user to the search screen. However the user can also finish the task from this screen and the system will then return to the task screen.

FIGURE 4.8: EXPLICIT FEEDBACK SCREEN

4.3.1.2. RELEVANCE FEEDBACK CAPTURING COMPONENT

This component was developed to capture the relevance feedback during the search process. This component is independent of, but not isolated from, the user interface; it works behind the user interface and captures the readings from the screens. As shown in *Fig 4.9*, the captured relevance feedback includes implicit parameters, explicit parameters and user queries. The implicit parameters include: document Id; document hyperlink; visit time stamp; time on page; number of mouse clicks; mouse movement; mouse scrolling; scroll bar holding; key down times; key up times; book mark; save and print. Explicitly, the users are asked to rate the visited documents indicating their relevance to the query. The users and their tasks are identified through their unique user IDs. The query information includes; query text; query time stamp and number of retrieved documents, based on the query.

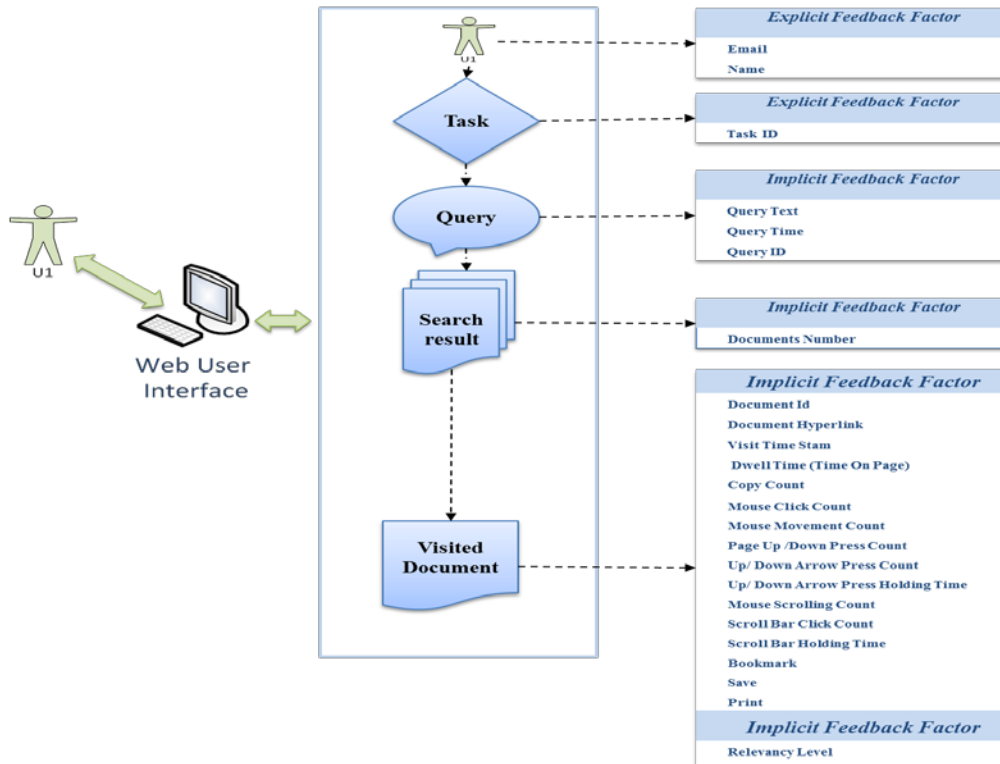


FIGURE 4.9: RELEVANCE FEEDBACK COLLECTION

4.3.1.3. SOLR COMPONENTS

Solr is a java-based open source enterprise-search server that operates under an Apache licence. Solr can communicate using XML and HTTP and it powers the search facility for many public websites such as CNet, Zappos, and Netflix, as well as intranet sites. In addition to the standard ability to return a list of search results for a query, it also has various other features such as, result highlighting; faceted navigation (for example, the ones found on most e-commerce sites); query spell correction; auto-suggest queries; and “more” and “like this” functions for finding similar documents.

Apache Lucene is the core technology underlying Solr. Lucene is an open source, high-performance text search library which could be used to build search server. (McCandless, Hatcher and Gospodnetic, 2010; Apache Lucene, 2014).

The following components of Solr were used:

Document Analysis Component: No search engine indexes text directly: rather, the text must be broken into a series of individual atomic elements called tokens. This is what happens during the “Analyse Document” step. Each token corresponds roughly to a “word” in the language, and this step determines how the textual fields in the document are divided into a series of tokens preparing to add it to the index.

Index Writer: After the input has been analysed, it's ready to be added to the index. Index-writer stores the input in a data structure known as an inverted index. The index is built incrementally of building blocks called segments, and each segment consists of a number of documents. In general, the Lucene index consists of the following main files:

1. Segments file: A single file contains the active segments information for each index. This file lists the segments by name, and it contains the size of each segment.
2. Fields information file: Documents in the index are composed of fields, and this file contains the information about the field.
3. Text information file: This core index file stores all of the terms and related information in the index, sorted by term.
4. Frequency file: This file contains the list of documents that contain the terms, along with the term frequency in each document.
5. Position file: This file contains the list of positions at which the term occurs within each document.

Query Parser: In order to process and search user's text query, it should be transformed into a common search syntax form which is called query object form. For example, the query may have Boolean operations, phrase queries (double quoted), wildcard terms, etc. for expressions that might need specific processing. Solr provides a package to build the query called Query Parser.

Result Renderer: This part is responsible for taking the search result as a plain list, and then puts it in the right order and finally passes it to the user interface.

4.3.1.4. TIKA COMPONENTS

Apache Tika is an open source document-acquiring and document-building server that is compatible with Solr since they are working under the Apache umbrella. It is able to acquire different types of documents using the standard library and more using the Plugins. (Fagin et al. 2003; Smiley and Pugh 2009) In the proposed framework, the document-acquiring and document-building component will be used from Tika.

Acquiring component: This component is responsible to extract the Metadata as well as the body of the document in a form of text, and then passes them to the document building component.

Document building component: The function of this component is to transform the document from stream-text to a field form such as: title; author; date of creation; body; link etc. This form will be understood by the Solr analysing component that will take it as input for indexing.

4.3.1.5. RELEVANCE FEEDBACK DATABASE

The captured feedback was stored in a relational database in order to provide an expressive representation of the data and the relationships between the data. For example, the use of a relational database enables the linking of users to: tier queries, completed search tasks and the documents they have visited. The database was designed to support the constructing of profiles in later stages, where every data point is reachable from: all other related data points; the visited document; the leading query(s); users; and the other documents which were visited for the same query, could be known. The relational database was implemented using oracle **11g** database. Oracle was selected as it is available free under the oracle university license, as well as providing strong text and data mining libraries which were used at later stages to index and search the information from user queries.

4.3.1.6. TEST COLLECTION INDEXING

After the system was implemented and deployed, the “TREC Enterprise 2007 Track” was used as the dataset to be used for the search. The dataset was uploaded to the cloud hosting service for indexing. The dataset was provided in the form of XML files where each file contains thousands of documents. This required a special parser to be written to allow Tika to recognise where each document starts and ends. The indexing process took 13 hours and 23 minutes. The index size on the disc was 1 GB. The functionality of the system was tested on a sample of queries. The purpose of this test was to make sure that there were no technical errors in the system, such as the lack of the ability to show the visited document in the right format, especially when using different web browsers.

4.4. DATA COLLECTION

As discussed in the literature review chapter, there are several relevance feedback parameters that should be captured and analysed as indicators of relevance. However, due to the limitations and the scope of the study such as the problem domain, available data set and technology, not all of these parameters were considered in this study. Recalling that the current study examined the relevance feedback in the enterprise search, puts the main focus on post-click parameters, dwell time and user query. Less significance was given to the click-through parameters, owing to the lack of document links in the user dataset. The lack of links in the documents, which is a common feature in enterprise documents, limited the amount and quality of the information that click-through parameters could have provided about the user behaviour. TABLE 4.2 describes the selected relevance feedback parameters that were captured in this user study.

TABLE 4.2: RELEVANCE FEEDBACK PARAMETERS' DESCRIPTION

<i>Level</i>	<i>Relevance Feedback Parameter</i>	<i>Parameter Description</i>	<i>Type</i>
Task	Task ID	The search task number	E
Query	Query ID	The automated unique Id the system gives to the query	I
	Query Text	The whole query text	I
	Timestamp	The date and time of the query	I
	Document number	Number of documents paper in search result for each query	I
Document Visit	Dwell Time (Time On Page)	The actual time that the user spent on the page. This means the time is only counted when the window is On and Focused.	I
	Copy Count	The number of the copy instances (if the user copied part of the text of the document).	I
	Mouse Click Count	The number of mouse clicks on the page whether they were on internal links or the other areas of the page	I
	Mouse Movement Count	The number of movements of the mouse on the page	I
	Page Up /Down Press Count	The number of times that these keys were pressed	I
	Up/ Down Arrow Press Count	The number of times that these keys were pressed	I
	Up/ Down Arrow Press Holding Time	The time spent holding these keys	I
	Mouse Scrolling Count	The number of mouse scrolls on page	I
	Scroll Bar Click Count	Number of the number of the clicks on the scroll bar	I
	Scroll Bar Holding Time	The time spent clicking scroll bar	I
	Bookmark	If the page was bookmarked	I
	Save	If the page was saved	I
	Print	If the page was printed	I
	Explicit Relevance Level	The user judgment on how relevant this page was their query. An integer number between 1 and 10	E
Category : (I) indicates Implicit and (E) indicates Explicit			

As discussed in section 4.2.1, the participants were given the search tasks to complete individually using their own queries. During the search process; their feedback was captured and stored in the relevance feedback database by the system. The data was captured based on the parameters described in Table 4.2. However, there was some other identification and supporting data captured, such as the user email address, which was used by the system to identify the user and the search session to indicate the search iteration. The system harvested 812 user queries and 1291 document visits which gave a reasonable size of relevance feedback for statistical analysis and profiling. Table 4.3 show a sample of the captured data. The data in the table is created by an SQL query to combine the data from different tables in one view.

TABLE 4.3: SAMPLE OF RELEVANCE FEEDBACK DATA

email	taskid	text	documenturl	queryTime	timeOnPageSec	mouseClickCount	mouseMovementCount	pageUpDownPressCount	upDownArrowPressCount	upDownArrowScrollingCount	scrollBarClickCount	scrollBarClickHoldingTimeSec	bookmark	save	print	explicitRelevancyLevel
aa7857@c	T18	ocean currents and conditions	CSIRO120/CSIRO120-12466668.html	41904.58	150	9	0	135	0	0	0	0	0	0	0	9
zvunca.mi	T13	cement production techniques	CSIRO148/CSIRO148-08225989.html	41904.57	150	2	8	117	0	0	0	0	0	0	0	9
ab7702@c	T3	protection from bushfires	CSIRO172/CSIRO172-05138254.html	41904.58	150	4	0	110	0	0	0	195	0	0	0	9
zvunca.mi	T8	scramjet technology research	CSIRO129/CSIRO129-12428763.html	41842.63	150	1	0	0	0	0	0	0	0	0	0	9
yahyawisc	T20	genome cancer	CSIRO064/CSIRO064-16300205.html	41835.58	150	10	19	86	0	0	0	0	0	0	0	9
ab3505@c	T7	CTFT carbon nanotube yarns	CSIRO135/CSIRO135-10700629.html	41841.62	150	1	0	8	0	3	0.25	23	0	0	0	9
zvunca.mi	T17	new drilling technologies	CSIRO082/CSIRO082-13878470.html	41900.65	150	10	0	25	0	0	0	0	0	0	0	9
zvunca.mi	T3	bushfire publication	CSIRO207/CSIRO207-02814027.html	41904.57	150	1	0	0	0	0	0	0	0	0	0	9
abbadia@	T2	the impact of climate change in au	CSIRO147/CSIRO147-05759775.html	41904.58	150	1	2	9	0	0	0	0	0	0	0	9
zvunca.mi	T15	safety telecollaboration	CSIRO187/CSIRO187-13324922.html	41835.56	150	5	4	25	0	0	0	0	0	0	0	9
abbadia@	T12	communion	CSIRO112/CSIRO112-10105813.html	41904.59	150	1	0	66	0	0	0	0	0	0	0	9
aa9235@c	T8	scramjet technology australia	CSIRO206/CSIRO206-07248293.html	41900.66	150	1	1	38	0	0	0	0	0	0	0	9
zvunca.mi	T14	sensor network	CSIRO265/CSIRO265-04679923.html	41835.57	150	5	0	23	0	0	0	34	0	0	0	9
e-tool@h	T6	Biomedical and Medical textiles	CSIRO100/CSIRO100-13449774.html	41842.63	150	3	0	11	0	0	0	0	0	0	0	9
zvunca.mi	T7	carbon nanotube projects CTFT	CSIRO137/CSIRO137-13489295.html	41835.56	150	8	0	9	0	0	0	0	0	0	0	9
yahyawisc	T10	Wellbeing Diet book	CSIRO069/CSIRO069-13730023.html	41904.6	150	1	0	20	0	0	0	0	0	0	0	9
aa7785@c	T1	Gene Technology on cotton	CSIRO197/CSIRO197-04594121.html	41835.95	150	7	1	21	0	0	0	0	0	0	0	9

4.5. CONCLUSION

In this chapter a user study was carried out on 35 users in order to capture adequate relevance feedback for constructing the required profiles. The captured data included implicit and explicit feedback parameters in addition to the user queries. The users were invited to complete simulated search tasks using their own queries and to interact freely with the provided search system. The search tasks were designed to simulate a real life enterprise search scenarios and were developed based on the

dataset (TREC Enterprise Trak 2007). The user study resulted in capturing 812 user queries and 1291 document visits which provide a reasonable size of relevance feedback for statistical analysis and profiling.

The next chapter discusses the proposed approach including its component, methods and implementation details. It also discusses how the captured data was used for creating profiling and training the models in the proposed approach.

5. CHAPTER 5: PROPOSED APPROACH

5.1. INTRODUCTION

As discussed in the previous chapter, relevance feedback was captured by conducting a user study with 35 users on 20 enterprise related search tasks. The captured relevance feedback containing implicit parameters, explicit parameters and the queries of users, provided adequate information for developing the proposed approach. This chapter discusses the proposed approach for the recommender system for enterprise search. The proposed approach provided a new mechanism for constructing and integrating a task, user and document profile into a unified index through the use of relevance feedback and fuzzy rule based summarisation.

The recommender system helped to search for relevant documents and experts (people). The system filtered out irrelevant information and thus helped to overcome the information overloading problem in the context of enterprise search. This led to a better quality of retrieved information and a saving of time and effort in searching for the right information in right time. The proposed approach was designed to integrate the implicit and explicit relevance feedback in order to provide more accurate information based on user behaviour, while handling the uncertainty and the subjectivity in the user relevance feedback by using a fuzzy logic based mechanism. It created the recommendation based on the user query and its semantic relationship with the task, user and document. Importantly, the proposed approach was an adaptive approach, as the fuzzy rules and the predictive model which were used to estimate the relevance, was derived from the captured data and continued to learn from the user behaviour.

The rest of the chapter is organised as follows: Section 5.2 describes the proposed approach and its various phases. This section consists of the following subsections: Section 5.2.1 discusses relevance feedback collection from users of the search including what data was captured and how it was captured. Section 5.2.2 discusses document relevancy prediction based on the developed linear predictive model. It also discusses how the model was developed using the correlation and regression analysis. Section 5.2.3 explains the Fuzzy based Task, User and document profiling process

and how these profiles were created and structured. Section 5.2.4 discusses the Fuzzy combined Weight calculation and UTWI creation, based on the fuzzy rules summarization approach. It also describes the iterative training and validation process for the applying of the rules summarization approach, in order to extract the best set of rules for combined weight calculation. Section 5.2.5 discusses how the recommendations were created based on the relevance between the user query and the relevant search task, user and document. Section 5.2.6 describes how the recommendations were presented to the user through a web based user interface. Section 5.4 discusses the implementation of the proposed system which provides a description of the lower level implementation of the system and consists of the following subsections: Section 5.3.1 which describes the linear predictive model implementation; Section 5.3.2 which describes the implantation of the fuzzy based profiles constructing components; Section 5.3.3 which explains how the UTWI was implemented; Section 5.3.4 which describes the implantation of the recommendation creation and finally, Section 5.4 which is the conclusion of the chapter.

5.2. PROPOSED APPROACH

The large amount of digital information available in enterprises often causes information-overload, significantly increasing the amount of time and cognitive resources needed to acquire relevant and accurate information. This creates an urgent need for developing intelligent approaches that could help provide better search result quality by omitting irrelevant information decreasing the information overloading and saving time and effort. In the enterprise context, people in similar roles are likely to have similar interests and so could be searching for similar information. The search history for people in similar roles and the documents they have previously selected could enable recommendations of documents and people to be made based on the relevance feedback given when searches are being performed by other users in that role. However, in different cases in enterprise search, using a pure recommending approach such as collaborative filtering, content based filtering or a combination of these (i.e. hybrid approach) is not enough to meet the information needs of the users. For example, if a customer relationship or call centre staff member needed to answer a specific query about a technical problem or financial issue and this staff member is not part of the technical or financial team, then that staff member will not be able to get to the required recommended document and people. A good approach would be to

provide to the staff member the required information for the query by finding the relevant role first and then creating the recommendation based on that role.

As shown in *Fig.5.1*, the proposed approach combines the traditional information retrieval process which retrieved the relevant items based on a given user query and the recommendation process. This then created the recommendations based on the similarity between the users (e.g as in collaborative filtering), and/or between the user and the items, as in the content based approach. The search process in the proposed approach started by retrieving the relevant search tasks for a given user query and then based on the retrieved search tasks, the relevant documents and experts are recommended. The system created the recommendations based on a hybrid recommendation approach which combined the collaborative filtering collaborative (CF) and content based (CB) recommendation approaches. It used collaborative filtering to create task and document profiles and it also used the document profile, in order to provide a description of the features for the content of the item.

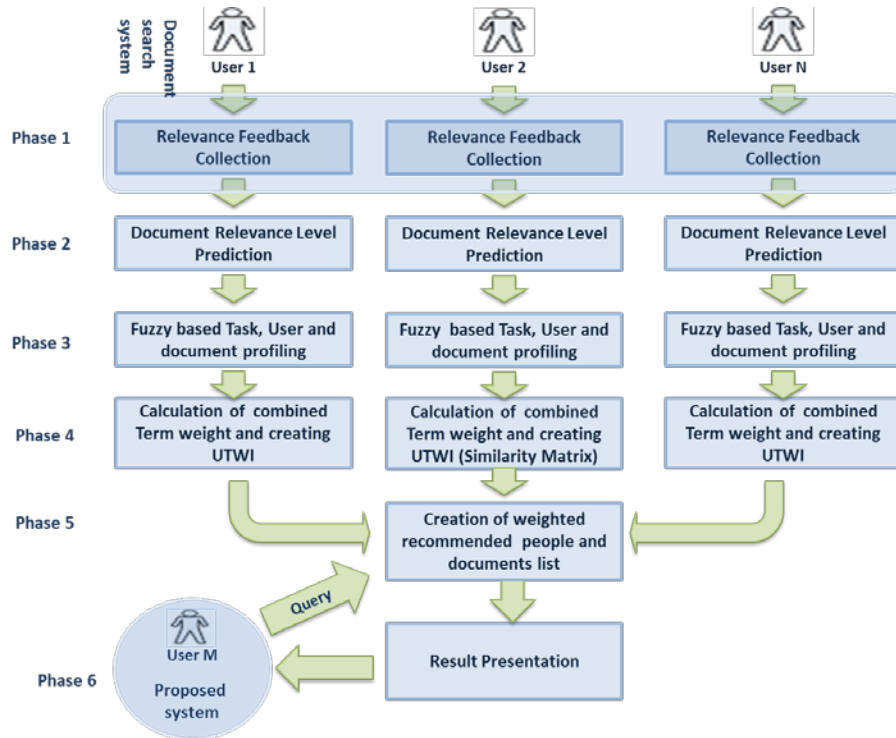


FIGURE 5.1 : PROPOSED APPROACH

It provided an intelligent adaptive and fuzzy based mechanism for recommending documents and people in an enterprise search context. In this mechanism, the relevance feedback from the user query was used as the main data source for developing a task, user and document profile. The task profile was modelled as a sequence of weighted

terms. The term weight reflected the relevance level of the term to the task, user and document profile. However, relevance feedback involved a high level of uncertainty owing to the inconsistency in user behaviour and subjectivity in their assessment of relevancy (Grzywaczewski & Iqbal, 2012). Therefore, handling such uncertainty was crucial to achieve better performance. A Fuzzy approach was used to overcome the uncertainty and bias in the user judgment in order to provide a normalized ranking method for recommendations in the enterprise search. The proposed approach consisted of six phases as shown in *Fig.5.1*.

The approach has the following advantages:

- **Uncertainty handling:** The fuzzy based mechanism helped to handle the uncertainty and the subjectivity of the user relevance feedback which provided a more realistic estimation of the relevance between the user query and the documents.
- **Semantic based recommendation:** The unified index enabled the creation of the recommendations based on the user query and the semantic relationship between the tasks, users and documents.
- **Implicit and explicit integration:** The proposed approach integrated the implicit and explicit relevance feedback parameters in order to produce a more accurate estimation of the document relevancy. However, to overcome the user unwillingness to provide explicit feedback, the approach captured it temporarily in order to retrain a predictive model in response to the changes in the data.
- **Adaptivity:** The approach was adaptive as the both linear predictive and the unified term weight calculation fuzzy rules were derived from the data and adapted according to significant changes in the common patterns within the data.

5.2.1. PHASE 1: RELEVANCE FEEDBACK COLLECTION

In this phase the relevance feedback was captured from the users during the search process. The captured relevance feedback included implicit parameters, explicit parameters and user queries. The implicit parameters included: document id, document hyperlink, visit time stamp, time on page, number of mouse clicks, mouse movement, mouse scrolling, scroll bar holding, key down times, key up times, book

mark, save, and print. Explicitly, the users were asked to rate the visited documents indicating their relevance to the query. The users and their tasks were identified through their unique user IDs. The query information included: query text, query time stamp and the number of retrieved documents based on the query.

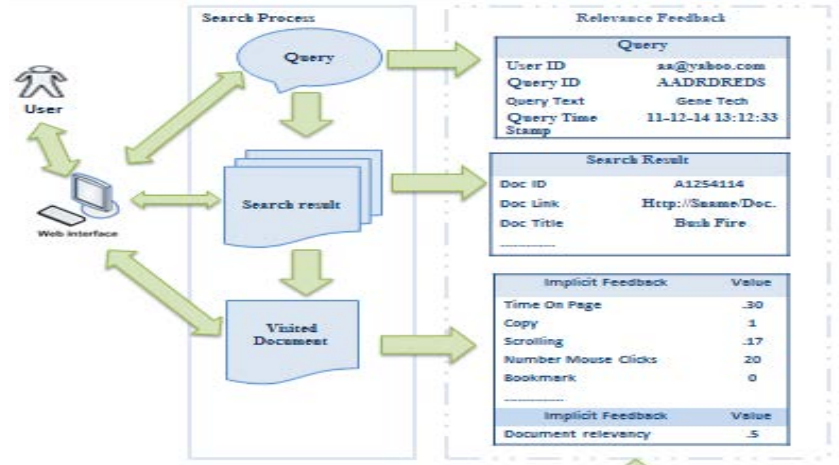


FIGURE 5.2 : RELEVANCE FEEDBACK COLLECTION

5.2.2. PHASE 2: DOCUMENT RELEVANCE PREDICTION

In this phase the relevance level of the visited document was predicted from the implicit feedback parameters. The predicted value was calculated by a linear predictive model which was developed using linear regression analysis. In linear regression analysis, regression models involved three types of parameters. The unknown parameters were also known as Coefficients (β), the independent variables were also known as Predictors (X), and the dependent variable was also known as Target (Y). A linear regression model related Y to a function of X and β (Mandel,2012)

$$Y \approx f(X, \beta) \quad (5.1)$$

The approximation is usually formalised as $E(Y | X) = f(X, \beta)$. In the more general multiple linear regression model with N independent variables and one dependent variable:

$$\hat{Y} = \beta_0 + \sum_{i=1}^N \beta_i X_i \quad .1 \leq i \leq N \quad (5.2)$$

Where \hat{Y} is the fitted predicted value of the dependent variable, β_0 is the intercept, β_i is the variable coefficient, X_i is the values of independent variable N is number of the independent variables.

In the approach, the model is developed using the following steps:

Step 1, Training Set Creation, in this step the training data set was created by associating the values of the implicit feedback parameters, which are discussed in phase 1, with the value of explicit relevance level for each document visit instance. As shown in Fig.5.3, the implicit and explicit parameters were categorised into independent variables (IV) or **Predictors** and dependent Variable (DV) or **Target**. The value of the predictors was used to estimate the value of the target.

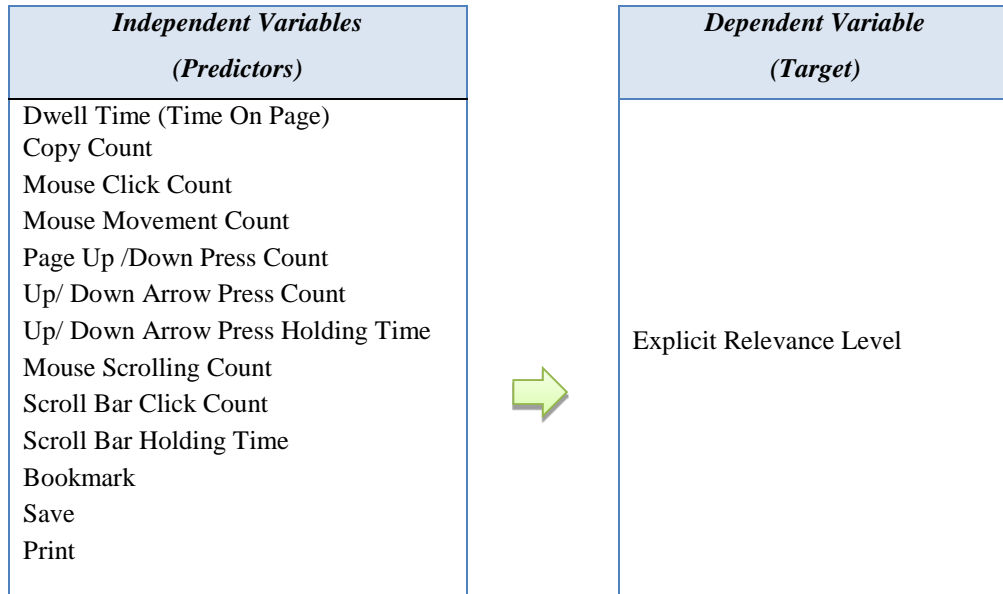


FIGURE 5.3 : PREDICTORS & TARGET

Step 2, Correlation Analysis, in this step the correlation analysis was carried out to extract only those independent variables (predictors) which have a significant correlation with the dependent variable (Target) and so considered for further analysis. Table 5.1 shows the correlation between the implicit parameters and the Explicit Relevance Level which was the explicit relevance feedback that the user was asked to provide. The value in each cell reflects the level of correlation between associated parameters and the signs: * and ** show the significance level of the correlation.

TABLE 5.1 : CORRELATION ANALYSIS

	Time On Page	Copy Count	Mouse Click Count	Mouse Movement Count	Page Up Down Press Count	Up Down Arrow Press Count	Up Down Arrow Press Holding Time	Mouse Scrolling Count	Scroll Bar Click Count	Scroll Bar Holding Time	Bookmark	Save	Print	Visit Time stamp	Explicit Search Satisfaction Level	Explicit Relevance Level
Time On Page	1	.019	.268**	.592**	.103**	.012	.019	.318**	.136**	.094**	.069*	-.024	.040	.025	.127**	.144**
Copy Count	.019	1	.165**	.017	-.004	.010	.023	.021	.137**	.292**	.047	.055	.154**	-.089**	.037	-.030
Mouse Click Count	.268**	.165**	1	.477**	.067*	.022	.037	.264**	.180**	.233**	.247**	.060	.193**	-.107**	.164**	.170**
Mouse Movement Count	.592**	.017	.477**	1	.060	-.030	-.026	.404**	.096**	.058	.148**	-.019	.054	-.013	.196**	.202**
Page Up Down Press Count	.103**	-.004	.067*	.060	1	-.005	-.005	.002	.135**	.073*	-.015	-.008	-.003	-.067*	-.011	-.007
Up Down Arrow Press Count	.012	.010	.022	-.030	-.005	1	.981**	-.024	.159**	.149**	-.023	-.014	.016	-.141**	-.038	-.032
Up Down Arrow Press Holding Time	.019	.023	.037	-.026	-.005	.981**	1	-.019	.191**	.186**	-.022	-.015	.033	-.150**	-.044	-.027
Mouse Scrolling Count	.318**	.021	.264**	.404**	.002	-.024	-.019	1	.045	.039	.139**	.012	.052	-.028	.172**	.189**
Scroll Bar Click Count	.136**	.137**	.180**	.096**	.135**	.159**	.191**	.045	1	.855**	.050	.042	.177**	-.079*	-.039	-.042
Scroll Bar Holding Time	.094**	.292**	.233**	.058	.073*	.149**	.186**	.039	.855**	1	.092**	.080**	.321**	-.079*	-.027	-.038
Bookmark	.069*	.047	.247**	.148**	-.015	-.023	-.022	.139**	.050	.092**	1	.333**	.094**	-.064*	.283**	.283**
Save	-.024	.055	.060	-.019	-.008	-.014	-.015	.012	.042	.080**	.333**	1	.202**	-.040	.136**	.143**
Print	.040	.154**	.193**	.054	-.003	.016	.033	.052	.177**	.321**	.094**	.202**	1	-.033	.024	.014
Explicit Relevance Level	.144**	-.030	.170**	.202**	-.007	-.032	-.027	.189**	-.042	-.038	.283**	.143**	.014	.083**	.945**	1
Visit Time stamp	.025	-.089**	-.107**	-.013	-.067*	-.141**	-.150**	-.028	-.079*	-.079*	-.064*	-.040	-.033	1	.077*	.083**
Explicit Search Result Satisfaction Level	.127**	.037	.164**	.196**	-.011	-.038	-.044	.172**	-.039	-.027	.283**	.136**	.024	.077*	1	.945**

** . Correlation is significant at the 0.01 level (2-tailed).

Correlation is significant at the 0.05 level (2-tailed).

The correlation analysis showed that candidate parameters for a linear relationship with the explicit relevance level of the document were: Dwell Time (Time on Page), Mouse Click Count, Mouse Movement Count, Mouse Scrolling Count, Save, Print and Visit Time Stamp. These parameters were considered in the regression analysis in the next step.

Step 3, Linear Predictive Model Development, in this step, the relevance level of the visited document was predicted from the implicit feedback parameters. The predicted value was calculated by the linear predictive model which was developed using the linear regression analysis.

The candidate parameters for a linear relationship with the explicit relevance level of the document were: Dwell Time (Time on Page), Mouse Click Count, Mouse Movement Count and Mouse Scrolling Count. The, Save, Print and Visit Time Stamp were considered **Predictors** and the explicit relevance level was considered as **Target** in the regression analysis. The regression analysis was carried out using IBM-SPSS-Statistics Version 22. Table 5.3 show the result of the analysis only the variables with a significance level $\geq .05$ were considered as predictors in the predictive model.

Table 5.2 Coefficients for the Target Explicit Relevance Feedback

TABLE 5.2 : COEFFICIENTS FOR THE TARGET EXPLICIT RELEVANCE FEEDBACK

<i>Model Term</i>		<i>Coefficient (β_i)</i>	<i>Sig</i>	<i>Importance</i>	
Intercept	(β_0)	1.395	.000	-	
Time on Page	(X_1)	.0069	(β_1)	.021	0.893
Mouse Scroll Count	(X_2)	0.013	(β_2)	.012	0.079
Mouse Movement Count	(X_3)	0.113	(β_3)	.031	0.028

Substituting the values form the table in *Equation (5.2)*, the predictive linear model for explicit relevance level becomes:

$$\hat{Y} = 1.395 + (X_1 \times .0069) + (X_2 \times .0069) + (X_3 \times .0069) \quad (5.3)$$

Then the importance of each predictor is used to normalize the value:

$$\hat{Y} = 1.395 + (X_1 \times .0069 \times .893) + (X_2 \times .0069 \times .079) + (X_3 \times .0069 \times .028) \quad (5.4)$$

Step 4, Model Validation, in this step the model is validated using R-squared R^2 method which is a common accuracy validating method for regression models β (Mandel, 2012). The validation method and results is discussed in more details in *Chapter 6*.

5.2.3. PHASE 3: FUZZY BASED TASK, USER AND DOCUMENT PROFILING

In this phase, three types of profiles were created: the search task profile, the user profile and the document profile. The profiles were created by employing an adaptive fuzzy logic approach. The approach was based on the method suggested by (Li & Kim, 2004) and modified to suit the query text analysis. The following sub sections describe the construction of the three profiles.

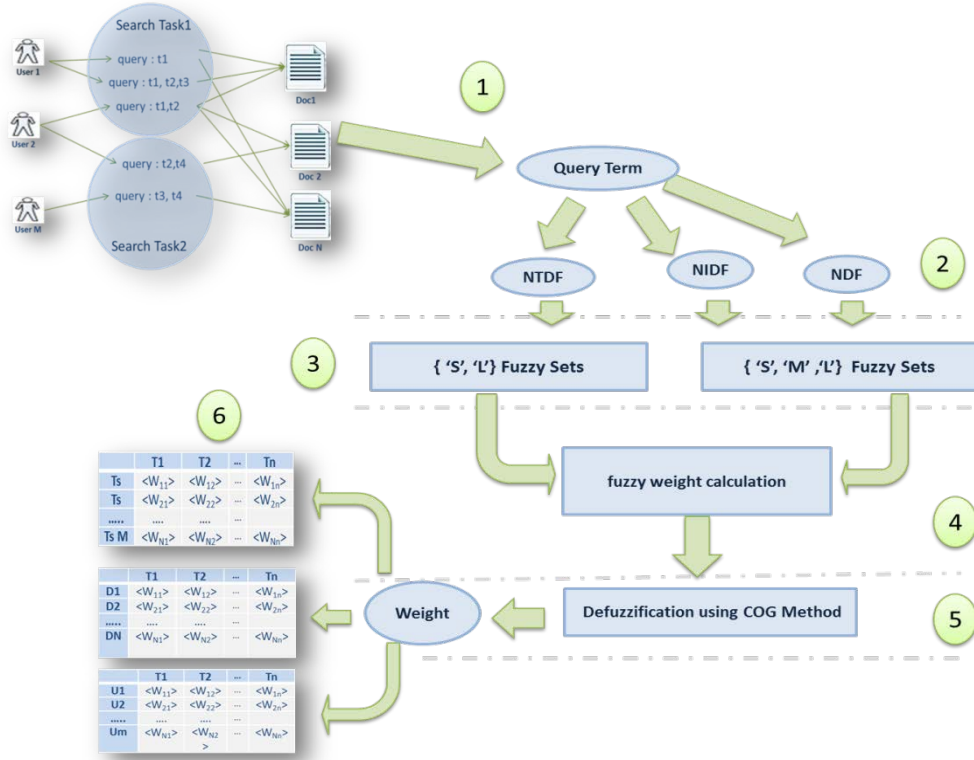


FIGURE 5.4 :FUZZY BASED TASK, USER AND DOCUMENT PROFILING

5.2.3.1. USER PROFILE

The user profiles were constructed using the information behaviour of users and their interest relevant to the search tasks they performed. In the profile each user is represented as a set of weighted terms which represent their interest. More precisely, the follow steps were followed to develop the user profile:

Step 1, A set of queries Q which led to document visits was selected.

Step 2, A set of all users U was selected and for each user $U_K \in U$ the following steps were carried out:

Step 3, a subset Ω_{U_k} of the query set Q is identified by selecting the queries that the user U_k has created.

Step 4, after identifying the sets Ω_{U_k} in step 3, the queries in the set were pre-processed and transformed into a set of candidate terms through eliminating stop-words and stemming by Porter's algorithm (Porter, 1980).

Step 5, the frequency measures: Distributed Term Frequency DTF, Document Frequency (DF), and Inverse Document Frequency (IDF) of each candidate term were calculated and normalized based on each set Ω_{U_k} and used as inputs to a fuzzy system

for calculating a weight for each term.

These frequency measures were used to calculate the term frequency in a document collection. However, they were also used in a collection of user queries where each user query could be considered as a document in order to calculate the frequency of the query terms (Poblete & Baeza-Yates, 2008). Based on that and in this step only, both terms: ‘document’ and ‘query’ refer to the user query. The DTF reflects the frequency and distributed status of a term in a set of user queries. This was calculated by dividing total occurrences of the term in the query set Ω_{U_k} by the number of the queries which contained the term in the set Ω_{U_k} . The DF represents the frequency of queries having a specific term within the set Q . The Normalized Document Frequency (NDF), is defined in Equation (5.5).

$$NDTF_i = \frac{\frac{TF_i}{DF_i}}{\text{Max}_j \left[\frac{TF_j}{DF_j} \right]} \quad (5.5)$$

Where, TF_i is the frequency of term t_i in the query set Ω_{U_k} , DF_i is the number of queries having term t_i in the query set Ω_{U_k} . i and $j = 1$ to M where M is the number of the terms in the set Ω_{U_k} . NDF, is defined in Equation (5.6).

$$NDF_i = \frac{DF_i}{\text{Max}_j DF_j} \quad (5.6)$$

Where; DF_i is the number of queries having term t_i in the in the query set Ω_{U_k} .

The IDF represents the frequency of the term in the query set Q rather than the set Ω_{U_k} . IDF was used to identify the terms which appear in many queries which might relate to different tasks, users and documents. These terms are not very useful for representing the relevance level and consequently they will be given a less weight than the others. The Normalized Inverse Document Frequency (NIDF) is defined as follows:

$$NIDF_i = \frac{IDF_i}{\text{Max}_j IDF_j}, \quad IDF_i = \text{Log} \frac{N}{n_i} \quad (5.7)$$

Where, N is the total number of queries in Q and n_i is the number of queries in Q in which the term t_i appears.

Step 6, in this step, the crisp values of the three input variables (NDTF, NDF, and NIDF) were fuzzified and mapped to sets of predefined fuzzy sets. As shown in Fig.

5.5. *a*, NDTF and NIDF have three linguistic labels { S(Small), M(Middle), L(Large) }, and NTDF as has two linguistic labels { S(Small), L(Large) }. As shown in Fig. 5.5. *b*, the output variable TW has six fuzzy sets associated with six linguistic labels { Z(Zero), S(Small), M(Middle), L(Large), X(Xlarge), XX(XXlarge) }.

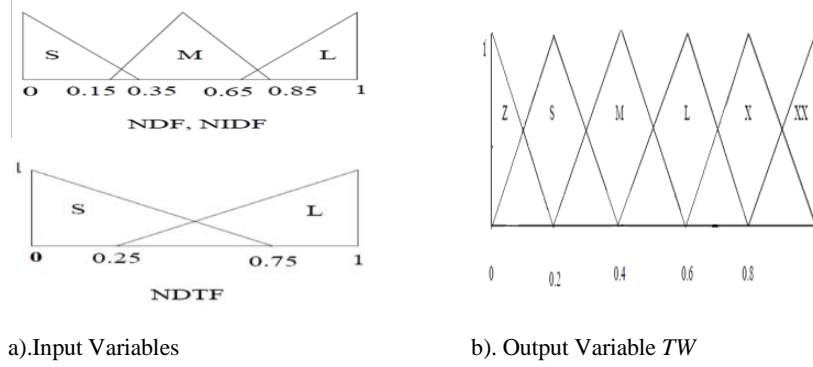


FIGURE 5.5: FUZZY SETS FOR INPUT VARIABLES

Step 7, in this step, the 18 ‘*If* \rightarrow *Then*’ fuzzy rules which are described by Ishibuchi and Yamamoto (2005) were used to infer a fuzzy term weight (TW) for the term t_i . These rules were constructed based on the assumption that the important or representative terms occurred across many queries in the representative query set Ω_{U_k} but not in the whole selected query set Q . In other words these terms have high NDF and NIDF values and low NDTF values. For example, as shown in Fig. 5.6, when NDF of a term is high and its NIDF is also high, the term is considered as a representative keyword so the output weight is between X and XX.

NIDF \ NDF	S	M	L
S	Z	Z	S
M	Z	M	L
L	S	L	X

NDTF = S

NIDF \ NDF	S	M	L
S	Z	S	M
M	Z	L	X
L	S	X	XX

NDTF = L

FIGURE 5.6 : WT CALCULATION FUZZY RULES

Step 8, the output of step 7, TW , was defuzzified using the center of gravity (COG) method in order get a crisp weight $TW_{U_k t_i}$ for each term to be added to the profile associated with the collection Ω_{U_k} .

Step 9, in this step the term t_i with its weight $TW_{U_k t_i}$ was added to the profile being created. However, as the system was used, more relevance feedback, including user

queries was captured and the system calculated new weights if the term frequencies changed. The profile was then updated by changing the term weight(s) to the new value(s).

More formally, let's assume P_{U_k} was the profile associated with the collection Ω_{U_k} and M was the number of terms in Ω_{U_k} then the profile P_{U_k} is defined as a set of weighted terms as follows:

$$P_{U_k} = \bigcup_{i=1}^M t_i TW_{U_k t_i} \quad (5.8)$$

As described in the previous steps, this phase resulted in the creation/updating of profiles for the users in the data set. The profiles were stored in a database table as shown in the Table 5.3.

TABLE 5.3 : SAMPLE OF USER PROFILE

<i>User ID</i>	<i>Term</i>	<i>Term Weight</i>
***569@coventry.ac.uk	OCEAN	0.571067496
***569@coventry.ac.uk	CURRENT	0.532861611
***569@coventry.ac.uk	CONDITION	0.50125059
***569@coventry.ac.uk	DIET	0.592570996
***569@coventry.ac.uk	ASSESSMENT	0.57609926
***569@coventry.ac.uk	PRODUCT	0.564558761
***569@coventry.ac.uk	LIFECYCLE	0.564558761

5.2.3.2. TASK PROFILE

In the task profile, each task was represented as a set of weighted terms that have been used to complete the search task. The weight reflected the relevance between each of these terms and the search tasks. The following steps were followed to construct task profile.

Step 1, a set of queries Q which led to document visits was selected.

Step 2, a set of all Tasks S was selected and each for each task $S_y \in S$ the following steps were carried out:

Step 3, a subset Ω_{S_y} of the query set Q is identified by selecting the user queries that were created to complete task S_y .

Step 4, after identifying the sets Ω_{S_y} in step 3, the queries in the set were pre-processed and transformed into a set of candidate terms through the same method as mentioned in

Step 4 of User profile.

Step 5, the frequency measures: Distributed Term Frequency DTF, Document Frequency (DF), and Inverse Document Frequency (IDF) of each candidate term were calculated and normalized based on each set Ω_{S_y} and used as inputs to a fuzzy system for calculating a weight for each term.

Step 6, in this step, the crisp values of the three input variables (NDTF, NDF, and NIDF) were fuzzified and mapped in the same way as in step 5 of user profile subsection.

Step 7, in this step, the 18 '*If* \rightarrow *Then*' fuzzy rules were used, which are described in step 7 of user profile constructing to infer a fuzzy term weight (*TW*) for the term t_i .

Step 8, the output of step 7, *TW*, was defuzzified using the center of gravity (COG) method in order get a crisp weight $TW_{S_y t_i}$ for each term to be added to the profile associated with the collection Ω_{S_y} .

Step 9, in this step the term t_i with its weight $TW_{S_y t_i}$ was added to the profile being created. However, as the system was used, more relevance feedback, including user queries was captured and the system calculated new weights if the term frequencies changed. The profile was then updated by changing the term weight(s) to the new value(s).

More formally, let's assume P_{S_y} was the profile associated with the collection Ω_{S_y} and M was the number of terms in Ω_{S_y} then the profile P_{S_y} was defined as a set of weighted terms as follows:

$$P_{S_y} = \bigcup_{i=1}^M t_i TW_{S_y t_i} \quad (5.9)$$

As described in the previous steps, this phase resulted in creating and updating profiles for the Tasks in the data set. The profiles were stored in a database table as shown in the Table 5.4.

TABLE 5.4 : SAMPLE TASK PROFILE

<i>Task Id</i>	<i>Term</i>	<i>Term Weight</i>
T1	TECHNOLOGY	0.363189988
T1	GENE	0.309313589
T1	RNAI	0.129074074
T1	MODIFY	0.094116972

T1	COTTON	0.090064157
T1	BIOTECHNOLOGY	0.088960647
T1	GENETICALLY	0.077359248
T1	FOCUS	0.039090293

5.2.3.3. DOCUMENT PROFILE

In the document profile, each document was represented as a set of weighted terms that were used by the users to retrieve the document relevant to their tasks. The weight reflected the relevance between each of these terms and the documents. The following steps were followed to construct document profile.

Step 1, a set of queries Q which led to document visits was selected.

Step 2, a set of all visited Documents D was selected and each for each task $D_g \in D$ the following steps were carried out:

Step 3, a subset Ω_{D_g} of the query set Q was identified by selecting the user queries that have led to visit the document D_g .

Step 4, after identifying the set Ω_{D_g} in step 3, the queries in the set were pre-processed and transformed into a set of candidate terms through the same method mentioned in Step 4 of User profile.

Step 5, the frequency measures: Distributed Term Frequency DTF, Document Frequency (DF), and Inverse Document Frequency (IDF) of each candidate term were calculated and normalized based on each set Ω_{D_g} and used as inputs to a fuzzy system for calculating a weight for each term.

Step 6, in this step, the crisp values of the three input variables (NDTF, NDF, and NIDF) are fuzzified and mapped in the same way in step 5 in user profile constructing.

Step 7, in this step, we used the 18 '*If \rightarrow Then*' fuzzy rules which are described in step 7 of user profile constructing to infer a fuzzy term weight (TW) for the term t_i .

Step 8, the output of step 7, TW , was defuzzified using the center of gravity (COG) method in order get a crisp weight $TW_{D_g t_i}$ for each term to be added to the profile associated with the collection Ω_{D_g} .

Step 9, in this step the term t_i with its weight $TW_{D_g t_i}$ was added to the profile being created. However, as the system was used, more relevance feedback, including user queries was captured and the system calculated new weights if the term frequencies changed. The profile will then be updated by changing the term weight(s) to the new

value(s).

More formally, let's assume P_{D_g} is the profile associated with the collection Ω_{D_g} and M is the number of terms in Ω_{D_g} then the profile P_{D_g} is defined as a set of weighted terms as follows:

$$P_{D_g} = \bigcup_{i=1}^M t_i T W_{D_g t_i} \quad (5.10)$$

As described in the previous steps, this phase results in creating/updating profiles for the documents in the data set. The profiles were stored in a database table as shown in the Table 5.5.

TABLE 5.5 : SAMPLE OF THE DOCUMENT PROFILE

<i>Doc URL</i>	<i>Term</i>	<i>Term Weight</i>
CSIRO000/CSIRO000-00000000.html	DIET	0.592570996
CSIRO000/CSIRO000-00000000.html	WELLBEING	0.520925064
CSIRO000/CSIRO000-00000000.html	TOTAL	0.511938243
CSIRO000/CSIRO000-00000000.html	DEVELOPMENT	0.5
CSIRO000/CSIRO000-00000000.html	DIETARY	0.5
CSIRO000/CSIRO000-00000000.html	TRIAL	0.5
CSIRO000/CSIRO000-14537203.html	HUMAN	0.592559456
CSIRO000/CSIRO000-14537203.html	CLINIC	0.53027557

5.2.4. PHASE 4: FUZZY COMBINED WEIGHT CALCULATION & UNIFIED TERM WEIGHT INDEX (UTWI) CREATION

In this phase the task, user and document profiles were combined in one index which was called the Unified Term Weight Index (UTWI) which was used as a similarity matrix for the recommender system. In this index each term had a unified weight per task, user and document. If term t_i was used by user U_k to retrieve the document D_g in order to complete the search task S_y then the unified term weight for t_i is W_{iykg} . This meant that the new weight considered the relevance between the term and the whole combination of the three factors; the user, the document and the task. This phase included the following steps:

Step 1, Fuzzy rules Extraction, let's say that V is the set of document visits in the data set which contains H visits, then V_h is the document visit instance where $h= 1$ to H . Each V_h is associated with the user query Q_e which led to this visit where $e=1$ to E and E is the number of queries in the dataset, the search task it occurred in S_y , the user

who made this visit U_k , the visited document D_g and the predicted relevance feedback R_h (see output of phase 2). Q_e consists of Z terms where t_{ez} is the query term in Q_e and $z = 1$ to Z . Then each t_{ez} is associated with its weight W in each of the profiles of S_y , U_k , and D_g that were computed in phase 3. These three weights are associated with the predicted relevance R_h . As a result, each term t_{ez} is represented as a set of four values $\{W_{s_y t_{mz}}, W_{u_k t_{mz}}, W_{d_g t_{mz}}, R_h\}$. If we consider the three first weights as inputs and the R_h as a result, then we have a sequence of three input values and one result value $\{W_{s_y t_{mz}}, W_{u_k t_{mz}}, W_{d_g t_{mz}} \rightarrow R_h\}$ for each instance in dataset.

The inputs and output values were mapped to predefined fuzzy sets with the linguistic labels ‘Low’ (L), ‘Medium’ (M) and ‘High’ (H) based on Mendel Wang method described in (Wu, Mendel, and Joo, 2010). As shown in Fig 5.7, in the proposed system, the shapes of the membership functions for each fuzzy set were based on triangle MFs.

A triangular MF is specified by three parameters $\{a, b, c\}$ as in Equation (5.11). Fig.5.7 illustrates a triangular MFs defined by triangle for the fuzzy sets.

$$Triangle(x; a; b; c) = \begin{cases} 0, & x \leq a. \\ \frac{x-a}{b-a}, & a \leq x \leq b. \\ \frac{c-x}{c-b}, & b \leq x \leq c. \\ 0, & c \leq x. \end{cases} \quad (5.11)$$

Where the parameters $\{a, b, c\}$ (with $a < b < c$) determine the x coordinates of the three corners of the underlying triangular MF.

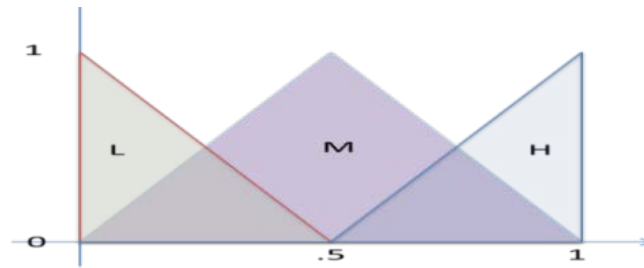


FIGURE 5.7 : FUZZY SETS FOR INPUT AND OUTPUT VARIABLES

The outcome from this step was a set of antecedents and consequents also called ‘if \rightarrow then’ fuzzy rules where each of the inputs and the outputs were represented by the associated linguistic label as shown in TABLE 5.6 If B is the linguistic label $\{‘L’, ‘M’, ‘H’\}$ of the value of each of the inputs and the output then the fuzzy rule

FR_h is:

$$B(W_{syt_{mz}}), B(W_{ukt_{mz}}), B(W_{dgt_{mz}}) \rightarrow B(R_h) \quad (5.12)$$

TABLE 5.6 : SAMPLE OF THE EXTRACTED FUZZY RULES

User	Task	Document	Term	W_u	W_T	W_D	\rightarrow	R
U1	T2	11884897.html	PUBLICATION	H	M	M	\rightarrow	L
U2	T20	09530858.html	PUBLICATION	M	M	M	\rightarrow	H
U3	T3	15292585.html	DIFFER	M	M	M	\rightarrow	M
U4	T3	15292585.html	SHEET	M	M	H	\rightarrow	M
U5	T3	15292585.html	TRIAL	M	M	M	\rightarrow	M
U3	T6	01314419.html	DIETARY	M	M	H	\rightarrow	L
U5	T2	01314419.html	TOTAL	L	M	M	\rightarrow	H
U6	T17	03452997.html	LEAD	M	M	H	\rightarrow	H
U7	T10	11659583.html	TOTAL	M	M	H	\rightarrow	L
U8	T6	03288103.html	CARBON	L	M	H	\rightarrow	M
U9	T2	13286918.html	FIRE	H	M	H	\rightarrow	L
U2	T20	01037988.html	COPPER	M	M	H	\rightarrow	M
U10	T17	04945868.html	CARBON	L	M	H	\rightarrow	H
U10	T17	04945868.html	PROJECT	L	M	M	\rightarrow	H
U11	T11	04945868.html	YARN	L	M	M	\rightarrow	L
U10	T17	04945868.html	YARN	M	M	M	\rightarrow	H
U6	T19	13878470.html	BITE	M	M	M	\rightarrow	L

Step 2, Best Fuzzy Rules Set Selection, in this step a K-fold of five folds ($k=5$) training and validation method was applied in order to select the rule set that achieves a high level of accuracy in classifying the relevance between the term and the document in a particular visit. This method will be discussed in details and numbers in Section 6.4 of the results and evaluation chapters. However, the applied method included the following sub steps:

Step 2.1, Dataset partitioning, in this step, the fuzzy rules set which resulted from *Step1* was partitioned into five equal sized folds and for each fold the following steps: (*Step 2.2*, *Step 2.2*, *Step 2.2*, *Step 2.2*) were carried out.

Step 2.2, Training rules set selection, in this step the training rules set was selected by holding out the current fold (k_j) The validation process was carried out in k iterations and in each iteration $j : 1$ to k the subset k_j was held out and called hold-out set D_h . The rest of the subsets were grouped in a training set $D_t = D - k_j$.

Step 2.2, Compression of Fuzzy Rules, in this step, a rule compression was performed on the fuzzy rules in the training set D_t . This was done in order to extract those rules with the maximum firing strength. The rule compression

technique was adopted from Wu, Mendel, and Joo, (2010), and used later by (Doctor and Iqbal 2012;Iqbal et al. 2014) It was based on two quality measures, namely generality and reliability, for each unique rule pattern. Generality refers to the number of instances representing the rule pattern (Wu, Mendel and Joo, 2010). Reliability reflects the confidence level in the rule pattern. Both generality and reliability were used to calculate the scaled weight of each unique rule pattern.

The rule generality was measured using a scaled fuzzy support. The scaled fuzzy support was a frequency ratio of a unique rule pattern in a set of rules having the same consequent as shown in *Equation (5.13)* and was based on the calculation described in (Ishibuchi & Yamamoto, 2005). In the proposed approach, it was used to identify and eliminate duplicate instances by compressing the rule base into a set of M unique and rules modelling the data.

$$scFuzzSup(\underline{FR}_l) = \frac{Co_{\underline{FR}_l}}{Co_{\underline{FR}_l} + Co_{\widehat{FR}_l}} \quad (5.13)$$

Where $l = 1$ to M , l is the index of the rule \underline{FR}_l is a unique antecedent combination associated with the consequent linguistic label B (unique rule pattern) and $Co_{\underline{FR}_l}$ is the number of instances which support the rule pattern \underline{FR}_l in the data set. \widehat{FR}_l is the set of the other antecedents combination which are different to \underline{FR}_l but have the same consequent as of \underline{FR}_l . $Co_{\widehat{FR}_l}$ is the number of the instances which support these other combinations \widehat{FR}_l .

The confidence of a rule is a measure of a rule's validity representing the strength of a unique rule pattern against contradictory rule patterns \widetilde{FR}_l which refer to the other rule patterns which have the same antecedent combination but with different consequent. Scaled rule confidence is calculated by taking the ratio between the number of instances representing a unique rule pattern $Co_{\underline{FR}_l}$ and the number of the instances representing the contradictory rules' patterns $Co_{\widetilde{FR}_l}$. *Equation (5.14)* and is based on the calculation described in (Ishibuchi & Yamamoto, 2005).

$$scConf(\underline{FR}_l) = \frac{Co_{\underline{FR}_l}}{Co_{\underline{FR}_l} + Co_{\widetilde{FR}_l}} \quad (5.14)$$

Step 2.3, Calculation of Scaled Rule Weights, in this step, the product of the scaled fuzzy support and confidence of a rule was used to calculate the rule's scaled fuzzy weight as shown in *Equation (5.15)*.

In this step, the unique rules' patterns resulting from the previous step were weighted by calculating a scaled fuzzy weight for each of the patterns. The scaled fuzzy weight was calculated as a multiplication of the scaled fuzzy support and the scaled confidence as shown in *Equation (5.15)* and described in (Ishibuchi & Yamamoto, 2005).

$$scWi = scFuzzSup \times scConf \quad (5.15)$$

The scaled fuzzy weight $scWi(s)$ are assigned to the generated M rules $scWi$ to take the following form:

$$B(W_{syt_{mz}}), B(W_{ukt_{mz}}), B(W_{dgt_{mz}}) \rightarrow B(R_h)[scWi] \quad (5.16)$$

The scaled fuzzy weight $scWi$ was used to measure how well the rule is able to represent the data. It was used to rank the fuzzy rule patterns in order to select the most representative rules as described in (Ishibuchi & Yamamoto, 2005). The scaled fuzzy rule weight was used to select rule patterns with the highest weights out of the other contradictory patterns. The selected rule patterns were used to build the fuzzy system in the next step.

Step 2.4, Rules classifying accuracy, in this step the extracted rules the resulted fuzzy rules were used to classify the relevancy of each instance in the hold-out data set D_h . The resulted relevancy classifications were compared with the associated linguistic labels of the predicted documented relevancy values from the linear predictive model. If the classification was identical the predicted relevancy linguistic label the comparing result is 1 otherwise 0. Then the average of the comparison represents the accuracy.

Step 2.4, Max Accuracy selection, in this step the rules set with maximum accuracy were selected to be used for building the fuzzy system which calculated the unified term weight in *step 4*.

Step 3, Calculation of the unified term weight: In this step the resulting rules from

the previous step were used to build a fuzzy system to calculate the unified term weight W_{iykg} for each query term t_i in each associated document visit V_h . The fuzzy system calculated the unified term weight based on the term weights in the profiles of the associated user U_k , document D_g and search task S_y which were created in phase 2 and the fuzzy rules extracted in step 3 of this phase.

The fuzzy system consisted of the three input variables $\{W_{s_y t_{mz}}, W_{u_k t_{mz}}, W_{d_g t_{mz}}\}$, one output variable which was the unified term weight W_{iykg} and the fuzzy rules which were extracted in step 3. The fuzzy system was then fed with the values of the inputs: $W_{s_y t_{mz}}$, $W_{u_k t_{mz}}$ and $W_{d_g t_{mz}}$ which were associated with each query term for each document visit in step 1. The calculated value of W_{iykg} was then used to create the UTWI which consists of $\{V_h, t_i, S_y, U_k, D_g, W_{iykg}\}$. UTWI was used in the next phase to create the recommendations.

5.2.5. PHASE 5: RECOMMENDATION OF DOCUMENTS AND PEOPLE (EXPERTS).

In this phase the recommender system's user query was pre-processed in the same way as in phase 1. The UTWI index was searched for the extracted query terms to find matching documents and people who visited those documents frequently, so called experts in that area. This started with finding the matching tasks in order to recommend documents and people based on the relevant task. A matching task should have at least one occurrence of at least one of the query terms in its associated instances in UTWI. Then for each of these tasks the average weight was calculated for each of the matching terms. Then these resulting weights were summed to give the aggregate task weight. Based on the aggregate task weight the relevant documents were extracted together with the users for each matching task. A relevant document/user should have at least one matching term occurrence in UTWI with the tasks terms. The average weight of each matching term was calculated for the relevant document/user and these were summed to calculate the aggregate weight of the relevant document/user. The document/users were sorted in descending order based on their aggregate weights.

5.2.6. PHASE 6: RECOMMENDATION PRESENTATION

In this phase, the recommended document and people were presented to the user through a web-based graphical user interface. The interface enabled the user to view the recommended documents as a weighted list in which each document was associated with its relevance weight for the user query. It also showed a user analysis chart in which the relevant people for the user query and their relevance weight were graphically represented. The user interface provided a query-task tree in which the relevant people (experts) were grouped based on tasks which were relevant to the user query.

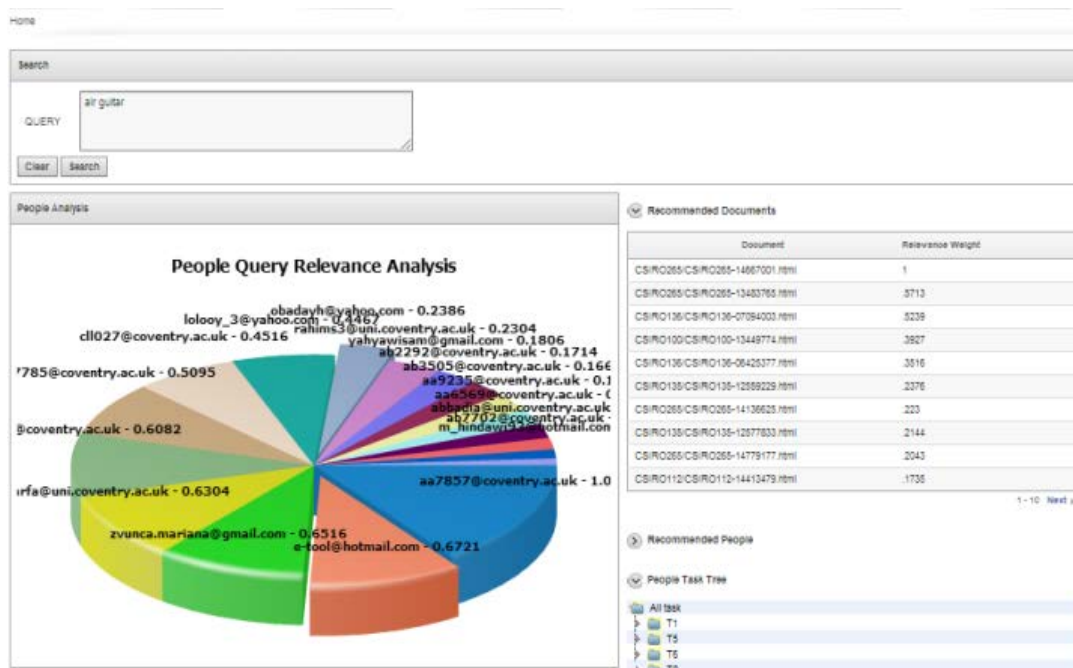


FIGURE 5.8: RECOMMENDER SYSTEM USER INTERFACE

5.3. IMPLEMENTATION

As shown in Fig 5.9, the proposed approach was implemented into a recommendation system to work as a Plugin or upper layer on the top of the enterprise search facility. In fact, it could be integrated with any search facility to recommend documents and people based on the users' search history. A prototype system was implemented in order to evaluate the accuracy of the proposed approach. The system was implemented using Java, Oracle 11g, Matlab fuzzy toolbox and IBM SPSS.

An Oracle Database was used to maintain the captured user feedback and the three profiles. Although relational databases have scalability issues it was used to implement a prototype system to test the accuracy of the model. As discussed in the

first chapter, the focus of this research is to improve the accuracy of the retrieved documents and therefore scalability and efficiency related to big data were not addressed.

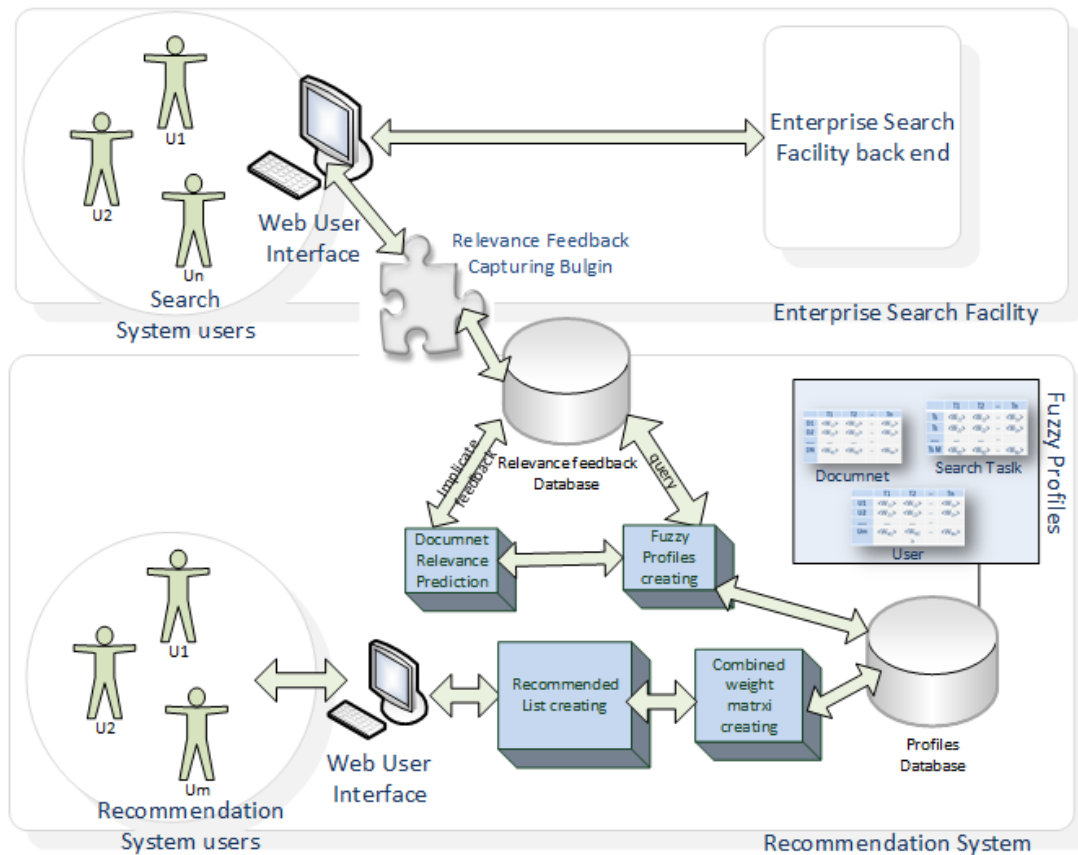


FIGURE 5.9: PROPOSED SYSTEM ARCHITECTURE

5.3.1. DOCUMENT RELEVANCE PREDICTION COMPONENT

This component function was based on the linear prediction model which was developed through the correlation and regression analysis as previously discussed. Both analyses were carried out using IBM SPSS by exporting the relevance feedback data from an Oracle database and importing it into IBM SPSS. The imported data was then analysed and the model equation was extracted. Finally the extracted equation was stored in the database and applied to the data for making the prediction. However, in the future this process could be fully automated by using IBM SPSS API's which will allow other applications to invoke the statistical engine of IBM SPSS or by using any other statistical Java library. The linear model equation was applied to the relevance feedback data to predict the document relevancy for each document visit in the database and the result was stored in database table.

5.3.2. FUZZY PROFILES CREATING COMPONENT

In order to create the profiles the user queries were stemmed and indexed using the Oracle text search library which allows the creation of an inverted index for any text record or file in the database. The users' queries were indexed to facilitate searching and also to extract the term frequencies. Then, Oracle SQL was used to create three database views to calculate NDF, IDF and NTFD for each term, based on the user, document and search task, as shown in Fig 5.10.

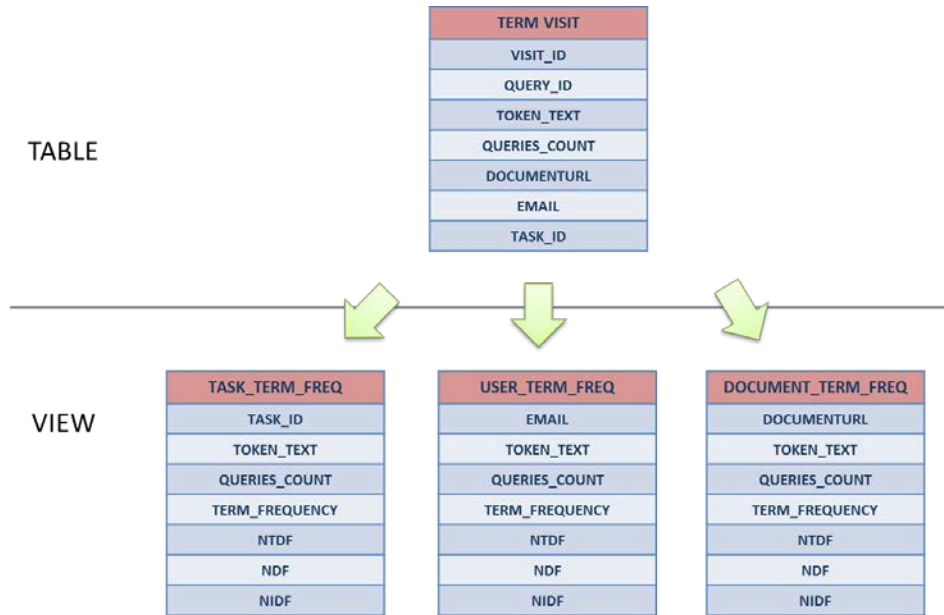


FIGURE 5.10 : TERMS FREQUENCIES VIEWS

The data from the three views were imported into the fuzzy controller (A) which was implemented using Matlab Fuzzy Logic Toolbox. The fuzzy system calculated the term weight based on the frequencies as described in Phase 3 of the proposed method. Fig 5.11 shows the fuzzy controller which consisted of the following:

- **Input variables:** NTFD, NDF and NIDF which are coloured in yellow.
- **Fuzzy rule base box (flc):** This contained the fuzzy rules as displayed in the front image of the same figure.
- **Output variable (WT):** This held the term weight for the associated profile.

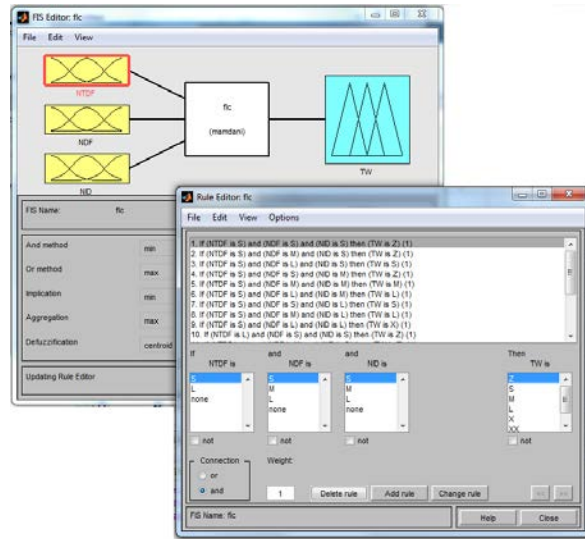


FIGURE 5.11: FUZZY CONTROLLER (A): PROFILE TERM WEIGHT

Fig 5.12 is a snapshot of the simulation of the controller behaviour based on the input parameters' values. The input variables; NTDF, NDF and NIDF appear in the first columns and yellow. The output variable TW appears in the fourth column and coloured in blue. The fuzzy rules in the rule base (*flc*) were applied to the values of the three input variables to calculate the value of the output variable TW.

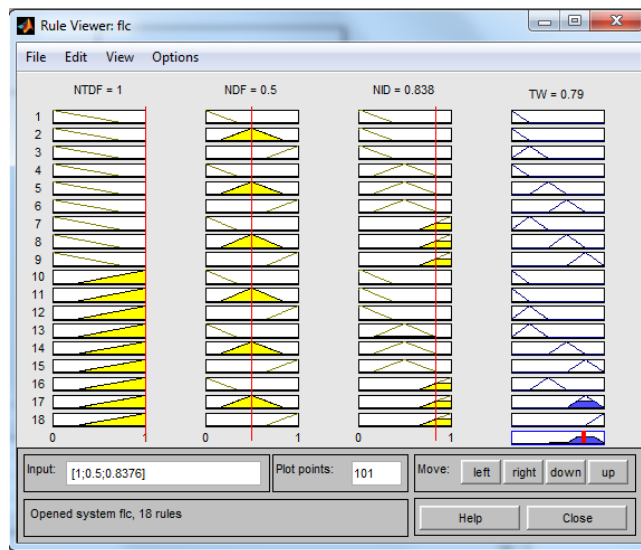


FIGURE 5.12: PROFILE TERM WEIGHT FUZZY CONTROLLER SIMULATION

The resulting TW values were stored in three database tables: USERS_PROFILES, TASK_PROFILES and DOCUMENT_PROFILES as discussed in Phase 3.

TASK_PROFILES	USER_PROFILES	DOCUMENT_PROFILES
TASK_ID	EMAIL	DOCUMENTURL
TOKEN_TEXT	TOKEN_TEXT	TOKEN_TEXT
T_T_WT	U_T_WT	D_T_WT

FIGURE 5.13 : PROFILES DATABASE TABLES

5.3.3. UNIFIED TERM WEIGHT INDEX (UTWI) CREATING COMPONENT

To implement this component Oracle SQL was used to create a view based on the term occurrence in the document visit to contain the term, the document visits, the term weight from the associated task, user and document profiles, and the predicted value of the document relevance for the document visits.

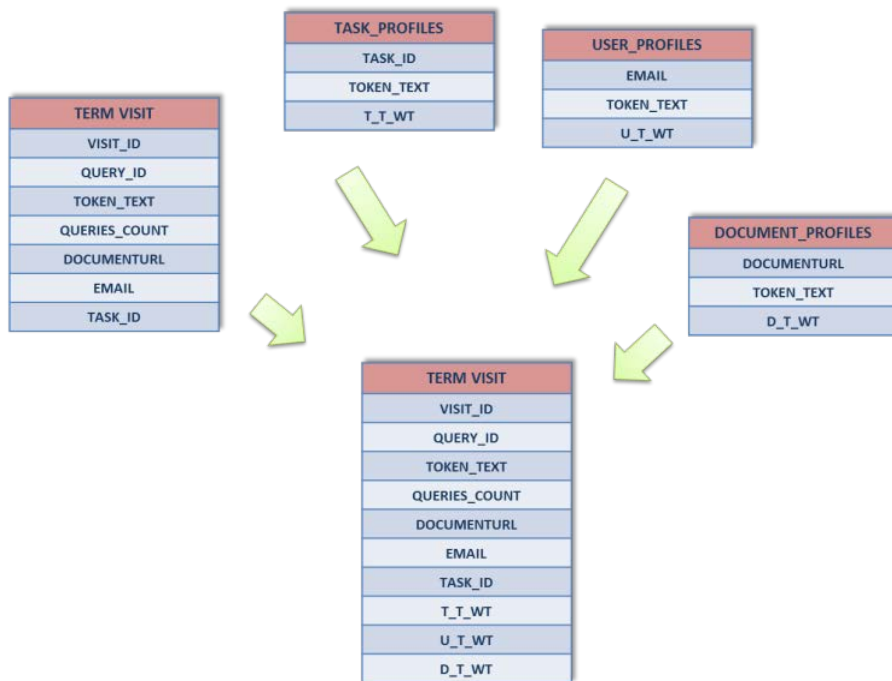


FIGURE 5.14 : TERM VISIT WEIGHTS

Then the term weights were imported into fuzzy set controller to convert them into linguistic label in order to convert theses weights into a form of fuzzy rules as discussed in Phase 5. The resulting rules were stored in a database table called ALL_RULES and based on the table and based on the database table a view called BEST_RULES was created which contains the rules with highest weights.

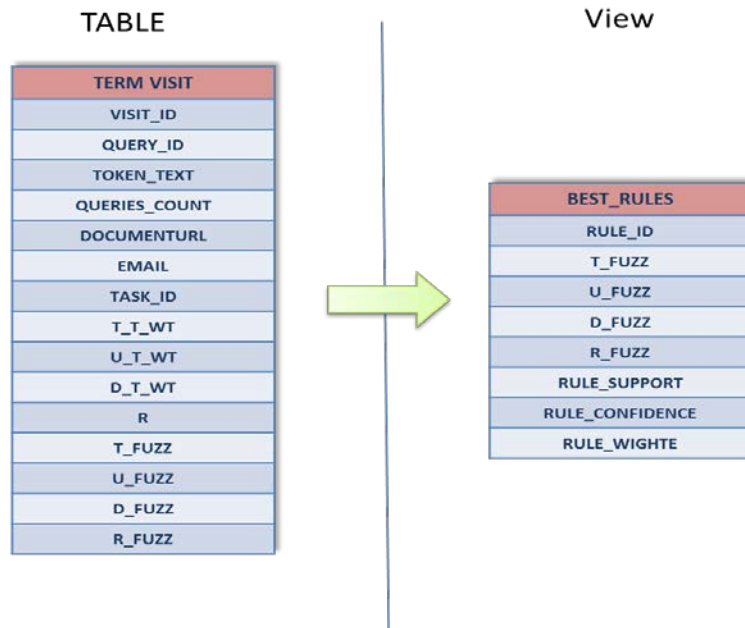


FIGURE 5.15 : BEST FUZZY RULES

These rules were used to build the fuzzy controller (**B**) in order to calculate the unified term weight which is shown in Fig 5.16. The Fuzzy controller (**B**) consists off the following:

- **Input variables TwT :** which was the weight of the term from the associated user's profile, **UwT** which was the weight of the term from the associated Task's profile and **DwT** which was the weight of the term from the associated Document's profile.
- **Fuzzy rule base box:** which contained the fuzzy rules from the database view BST_RULES. These rules were fired on the values of the three input valuables to give the value of the out variable.
- **Output variable $Unif_wT$:** which held the unified term weight.

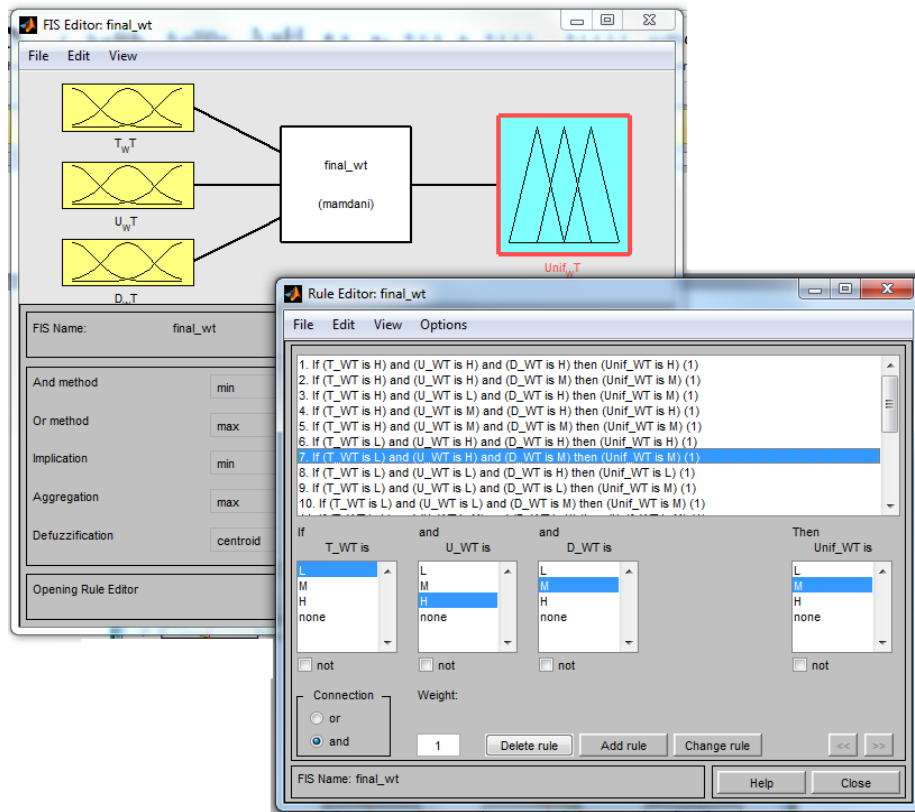


FIGURE 5.16: FUZZY CONTROLLER (B): UNIFIED TERM WEIGHT

Fig 5.17 shows a simulation for the controller behaviour based on the input parameters' values.

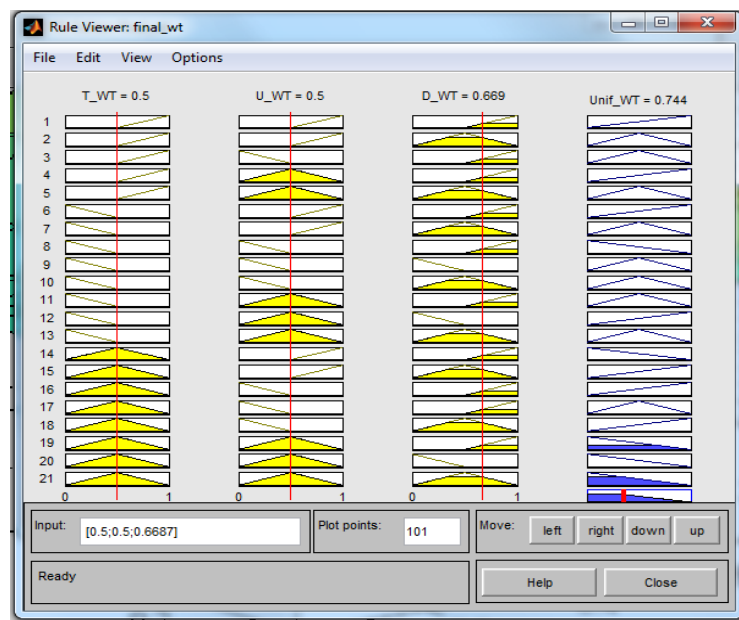


FIGURE 5.17: UNIFIED TERM WEIGHT FUZZY CONTROLLER

The resulting values of UnifwT stored in a database table called UTWI. Then based on the database table UTWI, two database views were created:

TASK_USER_WT and TASK_DOCUMENT_WT. The first one contained the user's weights for each task and the second one contained the documents' weights for each task. The weights were calculated accumulatively based on the weights of the terms that the users and the documents shared with the tasks.

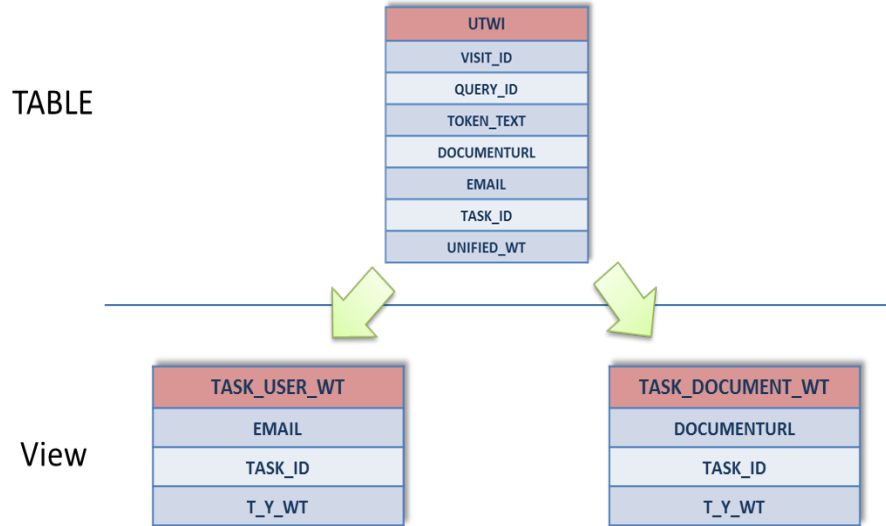


FIGURE 5.18: UTWI DATABASE TABLE. TASK_USER DATABASE VIEW AND TASK DOCUMENT DATABASE VIEW

5.3.4. RECOMMENDATIONS CREATING COMPONENT

This component was implemented by using Java web application developer for Oracle 11g. The component used the Oracle text search library to tokenise and index the user query and then passed the query to a stored PL/SQL procedure to find the most relevant search task for the user query based on the cumulative weights of the terms that the user query shares with each task, and the task with the highest weight. The recommended documents and people lists were created by a Select statement based on the relevant task.

```
SELECT c.TOKEN_TEXT , t.taskid , TASK_TERM_AC_WT , u.email , u.TASK_USER_AC
, TASK_TERM_AC_WT * u.TASK_USER_AC AS T_U_WT from
DR$CURRENT_QUERY_INDEX$I C, TASK_TERM_WT t , TASK_USER_WT u
where t.token_text = c.token_text and u.taskid = t.taskid
order by c.TOKEN_TEXT , t.taskid , TASK_TERM_AC_WT, u.TASK_USER_AC desc
```

FIGURE 5.19: SELECT STATEMENT FOR RECOMMENDED USER (EXPERT) LIST

5.4. CONCLUSION

In order to address the problem of information overload, this chapter has proposed a fuzzy based approach that provided a new mechanism for constructing and integrating three a task, user and document profile, into a unified index, through the use of relevance feedback and fuzzy rule based summarisation. The relevance feedback was used to develop a linear predictive model showing the association between the implicit and explicit feedback parameters. The model was used to predict the document relevancy from the implicit user feedback parameters. The predicted relevance values were used to identify the successful queries (which led to document visits) and train the fuzzy rule summarizing model. The successful user queries were pre-processed and the query terms were extracted. TF-IDF (Term Frequency and Inverse Document Frequency) matrices were calculated for the terms used by a fuzzy system to create the associated profiles (Task, User and Document). After the profiles were created, each term was then associated with its retrieved documents, the predicted relevancy and term weights corresponding to the task, user and document profiles. This formed a rule base consisting of three inputs (i.e. term weights associated with the three profiles) and one output, which was the predicted relevance level of the document. Then the fuzzy rule based summarisation was applied to extract the most representative fuzzy rules. These were used to build the unified relevancy index. A web-based user interface was developed to handle the user queries and enable recommending documents and people based on the user query.

The next chapter discusses the training and evaluation methods of the proposed approach. It presents the experimental results of the proposed system including the validation methods used such as R-squared and cross validation as well as the results of the comparative retrieval accuracy evaluation based Precision (**P**) and Recall (**R**).

6. CHAPTER 6: RESULTS AND EVALUATION

6.1. INTRODUCTION

The previous chapter discussed the proposed method and described in detail the method phases, starting with the capture of relevance feedback and ending with the presentation of results for the users. This chapter discusses the training and evaluation methods of the proposed approach. The proposed approach was evaluated at two levels: the accuracy of the component validation and the overall retrieval performance. The proposed system included two main components to estimate the relevance of the document, firstly the linear predictive model and secondly the fuzzy system, which was built based on the summarised fuzzy rules. The linear predictive model was validated using the R-squared method and the fuzzy system was validated using the K-Fold method. The overall retrieval accuracy of the proposed system was tested by carrying out a comparative retrieval accuracy evaluation in which the retrieval accuracy of the proposed system was compared with standard inverted index based Solr system and the semantic indexing based Lucid system. The comparative evaluation included two type of analysis: Precision (**P**) and Recall (**R**) analysis and document ranking analysis. Precision (**P**) and Recall (**R**) analysis were used to test the ability of the system to retrieve the relevant documents while the document ranking analysis was used to test the ability of the system to give a high ranking to the relevant document in the search result.

The rest of the chapter is organised as follows: Section 6.2 discusses the validation of the linear predictive. Section 6.3 discusses the summarised fuzzy rules training and validation including the results from the method used. Section 6.4 discusses the overall retrieval accuracy of the proposed system including Precision, Recall and document ranking categorical analysis. Finally, Section 6.5 concludes the chapter.

6.2. LINEAR PREDICTIVE MODEL VALIDATION

The model was validated using the R-squared (R^2) measurement. R-squared (R^2) measure is by far one of the most widely used and reported measures of the accuracy of statistical models (Mandel 2012). R^2 is formally calculated as shown in Equation 6.1.

$$R^2 = \frac{\sum_{i=1}^n (\hat{y} - \bar{y})^2}{\sum_{i=1}^n (y - \bar{y})^2} = \frac{SS \text{ Predicted}}{SS \text{ total}} \quad (6.1)$$

Where n is the number of the observation (data instances). \hat{y} is the predicted value of the data instance i , \bar{y} is the mean of the actual values of data instances in the set. y is the actual value of the data instance i . SS is sum of square.

As shown in Table 6.1, the accuracy of the predictive linear model was 76.5 %. The accuracy was calculated using the Equation 6.1. However, the accuracy can be improved through the adaptive mechanism of the proposed model by involving more document visits, users and search tasks.

TABLE 6.1: SUM SQUARES FOR THE LINEAR MODEL

Source	Sum of Squares	df	Mean Square	f
Predicted	9,510.566	5	1,902.173	747.940
Residuals	2,901.812	1,141	2.543	
Total	12,412.678	1,146		
Accuracy	76.5			

Fig 6.1, visualizes the linear relation between the estimated or predicted values and the observed or actual values of the dependent variable (explicit relevance level). The figure shows clearly that the pivot points between the predicted and actual values are almost distributed in a linear form which reflects the accuracy of the predictive model.

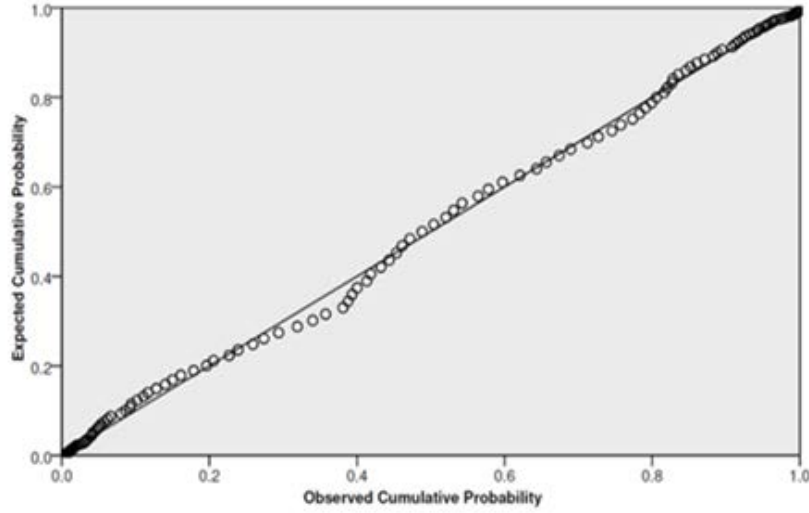


FIGURE 6.1 :PIVOT OF THE PREDICTED VALUE & ACTUAL VALUE

6.3. RULE BASED SUMMARISATION VALIDATION USING K-FOLD

The developed fuzzy rule based system was validated using a well-known validation method called k-fold cross-validation (Arlot and Celisse, 2010). In k-fold cross-validation, the dataset D is divided into k equal size (of size h items) subsets called folds. The validation process is carried out for k iterations and in each iteration $j : 1$ to k the subset k_j is held out and called hold-out set D_h . The rest of the subsets are grouped in a training set $D_t = D - k_j$. The accuracy of the model for each fold k_i was calculated using Equation 6.2.

$$acc_j = \frac{1}{h} \sum_{(v_i, y_i) \in D_h} \sigma(v_i, y_i) \quad (6.2)$$

Where $\sigma(v, y) = 1$ if $v=y$ and 0 otherwise. v_i is the predicted value of the instance i , y_i is the actual value of the instance i .

The final accuracy of the model is calculated by taking the average of the resulted accuracy values of the all iterations:

$$ACC = \frac{1}{h} \sum_{j=1}^k acc_j \quad (6.3)$$

In this method a dataset D is divided into two subsets; D_t (usually 80% of D) and D_h (usually 20% of D). Set D_t , named as the training set, was used to train the model, while D_h , called the testing set or hold-out set, is used to test the model in order to calculate the accuracy.

In order to train and validate the fuzzy rule based system a 5-fold cross validation was applied. The dataset was partitioned into five folds, representing 20% of the dataset. And then in each iteration one of these subsets was held out and the system was trained on the other folds representing the remaining 80% of the dataset to extract a set of weighted fuzzy rules. The resulting fuzzy rules of the training process were used to build a fuzzy system which classify the relevancy of each instance in the hold-out data. The resulting relevancy classifications were compared with the associated linguistic labels of the predicted documented relevancy values from the linear predictive model. Table 6.2 presents the resulting rule of the training process of the rule summarisation component for the first fold ($K=1$).

TABLE 6.2: SUMMARIZED WEIGHTED FUZZY RULES FOR $K=1$

	W _T	W _U	W _D		W _R	Firing Strength
1.	H	H	H	–	H	0.489292903
2.	H	H	M	–	H	0.364747958
3.	H	L	H	–	M	0.116836792
4.	H	L	M	–	M	0.082494544
5.	H	M	H	–	H	0.302517053
6.	H	M	L	–	M	0.010468469
7.	H	M	M	–	M	0.255007914
8.	L	H	H	–	M	0.120923391
9.	L	H	M	–	M	0.133836985
10.	L	L	H	–	M	0.034648234
11.	L	L	L	–	L	0.008734882
12.	L	L	M	–	L	0.020797199
13.	L	M	H	–	M	0.071606
14.	L	M	L	–	L	0.02148739
15.	L	M	M	–	L	0.057765699
16.	M	H	H	–	M	0.323666132
17.	M	H	L	–	L	0.003785873
18.	M	H	M	–	M	0.161731815
19.	M	L	H	–	M	0.084251856
20.	M	L	L	–	L	0.019945581
21.	M	L	M	–	M	0.053061553
22.	M	M	H	–	M	0.207267626
23.	M	M	L	–	L	0.048104422
24.	M	M	M	–	M	0.143720462

Table 6.3 presents a sample of the fuzzy classifier accuracy for the fold one (the first iteration). The system showed 86% accuracy performance in correctly classifying the relevance of the document.

TABLE 6.3: SAMPLE OF SUMMARISED FUZZY RULES ACCURACY

Term	Task	User	Document	E	A	C
Cotton	T13	**@coventry.ac.uk	04574741.html	M	M	1
Tech	T1	**@coventry.ac.uk	04587909.html	M	L	0
Air	T12	**@uni.coventry.ac.uk	04736857.html	M	M	1
Guitar	T13	**@yahoo.com	12228999.html	M	M	1
Bush	T13	**@coventry.ac.uk	12361974.html	M	M	1
Fire	T13	**@uni.coventry.ac.uk	14213537.html	M	M	1
Cotton	T12	**@coventry.ac.uk	16068064.html	M	M	1
Australia	T12	**@coventry.ac.uk	02493670.html	M	M	1
Australia	T13	**@gmail.com	03007618.html	M	M	1
School	T1	**@coventry.ac.uk	16400222.html	M	L	0
Cooper	T13	**@coventry.ac.uk	16400222.html	M	M	1
Cooper	T13	**@yahoo.com	16400222.html	M	M	1
Tech	T12	**@coventry.ac.uk	16456196.html	M	M	1
Bio	T12	**@yahoo.com	16456196.html	M	M	1
Lab	T13	**@gmail.com	08225989.html	M	M	1
.....
						86%

This process was repeated five times for the five different folds and for each fold the accuracy was calculated for the classifier, built based on the extracted rules. Table 6.4 presents the accuracy for each fold. It can be seen from the table that the highest (89%) accuracy was for the fold four at K=4 and the lowest accuracy (83%) was for the fold five at K=5. However, the average accuracy of the system is 86.2% which was relatively good as the initial performance, hence it was expected to improve as data was captured from more users.

TABLE 6.4: K-FOLD ACCURACY

<i>Fold (K)</i>	<i>Accuracy</i>
1	86%
2	88%
3	85%
4	89%
5	83%
Average	86.2%

In the proposed method the rules set with the highest accuracy was used in the fuzzy system to calculate the unified term weight in the UTWI which meant that the rule set associated with the fold 4 which is shown in Table 6.5 was used. Comparing the set of rules in Table 6.2 which was extracted from fold 1, with the set in Table 6.5 which was extracted from fold 4, it can be seen that there are three rules: 17, 21 and 22 which have changed while the other rules have stayed the same.

TABLE 6.5: SUMMARIZED WEIGHTED FUZZY RULES FOR K=4

	W _T	W _U	W _D		W _R	Firing Strength
1.	H	H	H	–	H	0.489292903
2.	H	H	M	–	H	0.364747958
3.	H	L	H	–	M	0.116836792
4.	H	L	M	–	M	0.082494544
5.	H	M	H	–	H	0.302517053
6.	H	M	L	–	M	0.010468469
7.	H	M	M	–	M	0.255007914
8.	L	H	H	–	M	0.120923391
9.	L	H	M	–	M	0.133836985
10.	L	L	H	–	M	0.034648234
11.	L	L	L	–	L	0.008734882
12.	L	L	M	–	L	0.020797199
13.	L	M	H	–	L	0.071606
14.	L	M	L	–	L	0.02148739
15.	L	M	M	–	L	0.057765699
16.	M	H	H	–	M	0.323666132
17.	M	H	L	–	M	0.082212525
18.	M	H	M	–	M	0.161731815
19.	M	L	H	–	M	0.084251856
20.	M	L	L	–	L	0.019945581
21.	M	L	M	–	L	0.043096711
22.	M	M	H	–	H	0.108767889
23.	M	M	L	–	L	0.048104422
24.	M	M	M	–	M	0.143720462

6.4. EVALUATION USING PRECISION, RECALL AND RANKING ANALYSIS

In order to evaluate the overall retrieval accuracy of the proposed system; a comparative evaluation was conducted to compare the retrieval accuracy measures of the proposed system with the existing standard search system and the semantic based enterprise search Lucid. Both systems were based on Solr as a core search platform, however, the first one used the standard inverted index while the second used semantic indexing (LucidWorks 2015). The standard inverted index consisted of the terms frequencies in the indexed documents. In semantic indexing the terms were given semantic weights to reflect their relevance to the indexed documents. The evaluation included Precision (**P**) and Recall (**R**) analysis and document ranking analysis. Precision (**P**) and Recall (**R**) were used to test the ability of the system to retrieve the relevant documents while the document ranking analysis was used to test the ability of the system to give a high ranking to the relevant document in the search result.

6.4.1. PRECISION AND RECALL ANALYSIS

The comparative retrieval accuracy evaluation was based on Precision (**P**) and Recall (**R**). In the comparative accuracy evaluation the values of **P** and **R** for the existing standard search system and the proposed recommender system were compared. Precision (**P**) and Recall (**R**) were standard evaluation metrics used in information retrieval research (Kelly, 2008). Precision (**P**) was a measure of the ability of a system to present only relevant items. **P** was defined as follows:

$$P = \frac{|Ra|}{|A|} \quad (6.4)$$

Where **Ra** was the number of relevant items retrieved and **A** was the total number of items retrieved in response to a user query.

Recall (**R**) was a measure of the ability of a system to present all relevant items. **R** was defined as follows:

$$R = \frac{|Ra|}{|Rm|} \quad (6.5)$$

Where ***Ra*** was the number relevant items retrieved and ***Rm*** was the total number of items retrieved.

The evaluation was carried out using the first 25 of the provided queries in the labelled data (TREC test collection) as these were used for creating the simulated search tasks in the user study. The results are shown in Table 6.6 . The queries were given to the three search systems: Standard Solr, Lucid and the proposed system and ***P*** and ***R*** were calculated for each of the given queries for each system. Then the averages ***P*** and ***R*** were calculated for each system.

TABLE 6.6: PRECISION (P) AND RECALL (R) FOR: STANDARD VECTOR SPACE SEARCH SYSTEM (STD SOLR), SEMANTIC BASED SEARCH SYSTEM (LUCID SOLR) AND THE PROPOSED RECOMMENDER SYSTEM.

QUERY ID	Precision(P)			Recall (R)		
	Std Solr	Lucid Solr	Proposed	Std Solr	Lucid Solr	Proposed
CE-001	0	0.006	0.023	0	0.5	0.667
CE-002	0.029	0.004	0.036	0.667	1	0.667
CE-003	0	0.005	0.061	0	1	1
CE-004	0	0.063	0.063	0	0.333	0.667
CE-005	0	0.007	0.035	0	0.5	0.667
CE-006	0	0.006	0.097	0	0.75	0.75
CE-007	0.006	0.105	0.105	0.2	0.727	0.727
CE-008	0.004	0.023	0.188	0.308	0.692	0.692
CE-009	0.003	0.163	0.163	0.1	0.7	0.7
CE-010	0.007	0.041	0.041	0.5	1	1
CE-011	0	0.008	0.008	0	1	1
CE-012	0.001	0.057	0.057	0.333	0.5	0.5
CE-013	0	0.02	0.02	0	0.333	0.333
CE-014	0.004	0.023	0.023	1	0.333	1
CE-015	0.004	0.012	0.019	1	1	1
CE-016	0.022	0.002	0.036	1	1	1
CE-017	0	0.003	0.015	0	1	1
CE-018	0	0.007	0.038	1	1	1
CE-019	0	0.009	0.049	0	0.667	0.5
CE-020	0.004	0.003	0.036	1	1	1
CE-021	0.003	0.008	0.12	0.667	1	1
CE-022	0.014	0.031	0.075	0.667	0.333	1
CE-023	0.001	0.033	0.094	1	1	1
CE-024	0.005	0.035	0.063	0.8	1	1
CE-025	0	0.071	0.132	0.667	0.833	0.833
AVG	0.00428	0.0298	0.06388	0.43636	0.76804	0.82812

As shown in Fig .6.2, the average P value for a standard Solr system was 0.00428 which is relatively low. This low value of P indicated that the system retrieved a large number of irrelevant documents. The P value for the semantic based search Lucid was 0.0298 which was significantly higher but still indicated a large number of retrieved documents. The proposed approach enhanced the value of P significantly where the average of P value increased 0.064. In other words, the proposed approach reduced the number of irrelevant documents in the search result which meant that the ability of the system to show only the relevant documents was enhanced. Referring to Equation 6.4, we find that the increment of P can either be due to the increase in the number of relevant documents retrieved or because of a decrease in the number of non-relevant documents retrieved or both of these factors. In the proposed approach, the number of the relevant retrieved documents increased and this was clear as the value of R has increased. Also, the number of non-relevant document had decreased significantly because the recommended or retrieved document list was filtered based on the fuzzy rule based profiling.

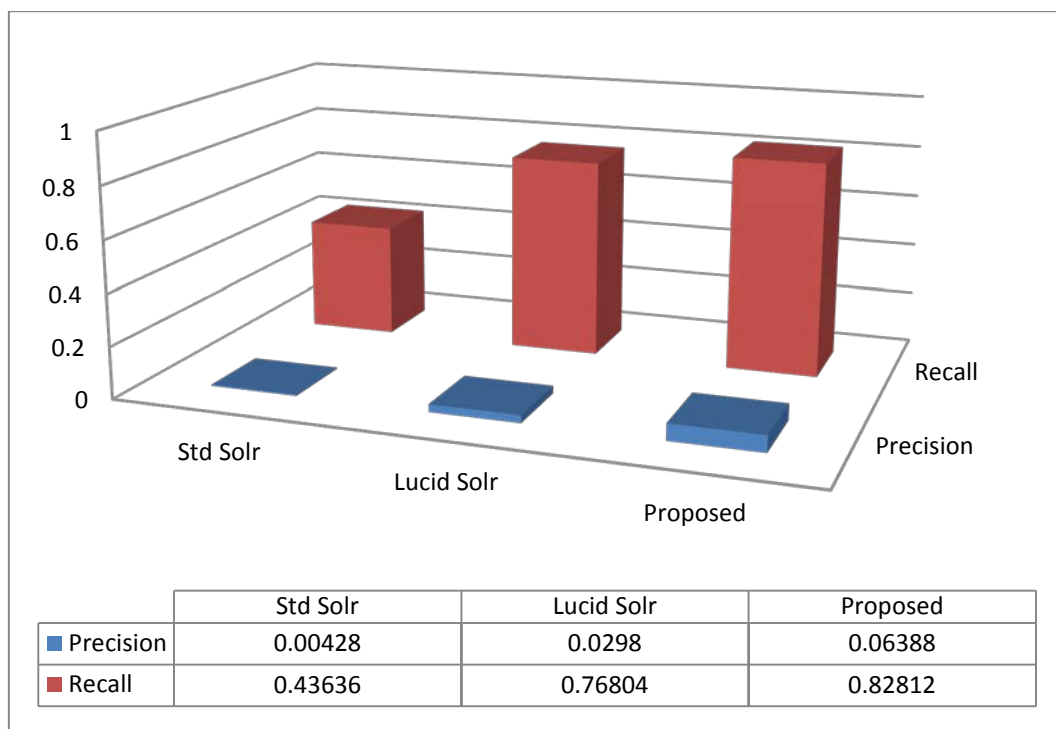


FIGURE 6.2:PRECISION (P) AND RECALL (R) FOR: STANDARD VECTOR SPACE SEARCH SYSTEM (STD SOLR), SEMANTIC BASED SEARCH SYSTEM (LUCID SOLR) AND THE PROPOSED RECOMMENDER SYSTEM

The proposed system has also enhanced the value of R . As shown clearly in Fig .6.2, the average value of R has increased significantly. Compared the standard Solr the proposed system enhanced the R value from 0.436 to 0.828 which meant the

ability of the system to retrieve the relevant document was enhanced as the proposed system produced results based on the user query and the semantic relationship among the tasks, users and documents. The proposed system outperformed the semantic based search system as well, but with a smaller difference as the value of R was enhanced from 0.76804) to 0.828 .

6.4.2. COMPARATIVE DOCUMENT RANKING ANALYSIS

The search system was not only required to retrieve the relevant document, but also to show it at the top of the search result (Agichtein 2006; Collins-Thompson 2011). Research shows that search engines try to show the relevant information in the first ten results to attract user attention (Wu 2009). Precision and recall ratios do not indicate the ranking of the document or whether or not the document was shown in the first 10 documents. Therefore, a comparative document ranking analysis was carried out to find out if the proposed approach improved the ability of the system to give a high ranking to the retrieved relevant documents and to push them to the top of the search list. In this analysis, the ranking of the relevant documents for the first 25 queries, which were used to create the search tasks for data capturing, was divided into 7 categories which were: A (1-5), B (6-10) , C(11 -20), D (21-30), E(30-40), F (> 40) and G (Not retrieved) and then the number of relevant documents which fell into each category was calculated. These frequencies were then compared to the frequencies resulting from the standard Solr and Lucid search systems for the same set of queries as shown in Table 6.7

TABLE 6.7: COMPARED DOCUMENT FREQUENCIES FOR RANK CATEGORIES

<i>Ranking Category</i>	Number of relevant documents			Percentage of relevant document		
	<i>Std Solr</i>	<i>Lucid Sor</i>	<i>Proposed</i>	<i>Std Solr</i>	<i>Lucid Sor</i>	<i>Proposed</i>
A (1 – 10)	15	50	60	15.96%	53.19%	63.83%
B (11 – 20)	13	13	14	13.83%	13.83%	14.89%
C (21 – 30)	4	7	2	4.26%	7.45%	2.13%
D (31- 40)	7	2	1	7.45%	2.13%	1.06%
E (> 41)	14	1	1	14.89%	1.06%	1.06%
F (Not Retrieved)	41	21	16	43.62%	22.34%	17.02%
Total	94	94	94	100%	100%	100%

As shown in Fig 6.3, the proposed approach improved the systems ranking performance. Compared to the standard Solr, the percentage of documents ranked in category A increased from 15.96% to 63.83%. Also, the percentage of documents in

category D and E dropped down significantly from 7.45% to 1.06% and from 14.89% to 1.06%. Finally the category percentage decreased from 43.62% to 17.02%. The proposed system achieved a better ranking performance than the semantic based search Lucid where the system showed more relevant documents in category A and B than Lucid.

The improvement in the relevant document ranking was due to the user relevance feedback which was used to train and validate the system which together with the help of the user judgement pushed the relevant documents to the top of the list.

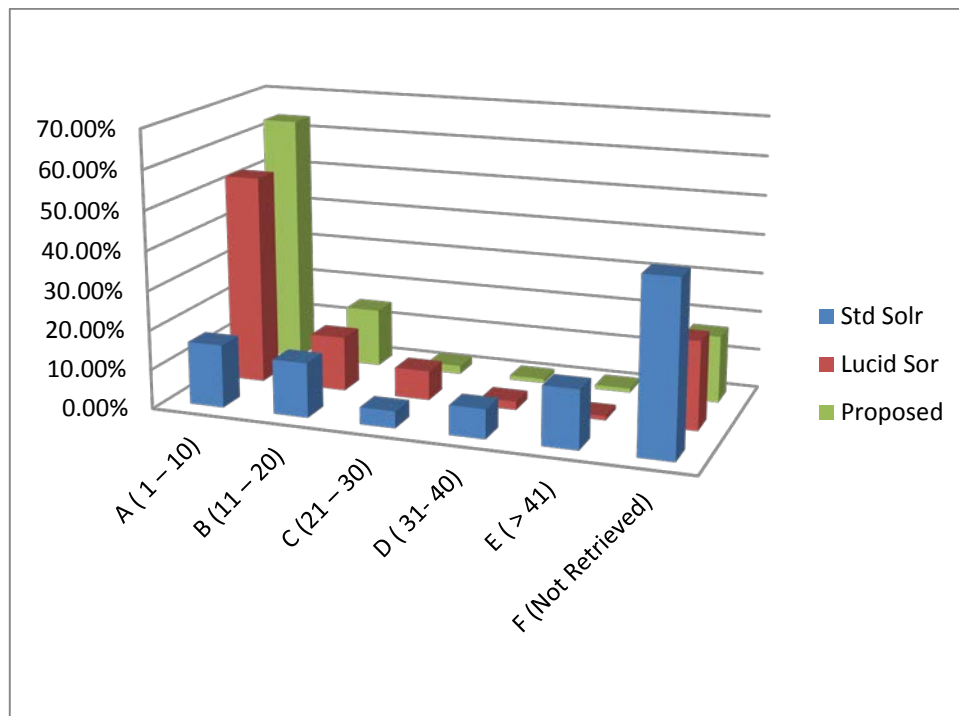


FIGURE 6.3: COMPARED DOCUMENT FREQUENCIES FOR RANK CATEGORIES

6.5. CONCLUSION

In this chapter, the proposed system was evaluated based on different measures and methods. The evaluation considered both the accuracy of the component and the overall retrieval performance. The accuracy performance of the linear predictive model was calculated using the R-squared method and the model showed 76.5 % accuracy in predicting the document relevance.

A 5-fold cross validation method was applied to train and validate the fuzzy classifier, which was built based on the summarised fuzzy rules. The highest accuracy performance that the classifier showed in classifying the documents-terms relevance was 89% at the fold-4 and the average accuracy for all folds was 86.2% which was

relatively good as an initial performance and was expected to improve as more data is captured from more users. The overall retrieval accuracy of the proposed system was tested by carrying out a comparative retrieval accuracy evaluation based on Precision (P), Recall (R) and ranking analysis. The values of P and R of the proposed system were compared to two other systems; Standard inverted index based Solr system and the semantic indexing based Lucid system. The proposed system enhanced the value of P significantly where the average of P value increased from 0.00428 to 0.064 compared with standard Solr system and from 0.0298 to 0.064 compared with Lucid. In other words, the proposed approach managed to decrease the number of irrelevant documents in the search result which means that the ability of the system to only show the relevant document was enhanced.

The proposed system enhanced the value of R as well. The average value of R was increased significantly (doubling) from (0.436) to (0.828) compared with standard Solr and from (0.76804) to (0.828) compared with Lucid. This meant that the ability of the system to retrieve the relevant document was also enhanced. Furthermore the ability of the system to rank higher the relevant documents improved compared with the other two systems.

As discussed previously, the proposed approach enhanced the values of both Recall and precision significantly which indicated that the number of the unwanted (i.e. irrelevant) documents was being minimised. This helped to overcome information overload by filtering out the unwanted information while maintaining a good retrieval accuracy performance. In addition the approach enhanced the ability of the system to show the relevant documents at the top of the search result page thus reducing the time consumed by the user to access the required information and in turn helped to address the problem of information overload. This tended to help the user spending the available time reading the relevant documents rather than wasting time searching for them.

The next chapter discusses the main conclusion of the thesis including the summary, contribution and limitations of the research. It also discusses the future research directions.

7. CHAPTER 7: CONCLUSION

7.1. INTRODUCTION

The previous chapter discussed the results of this research including the evaluation and the validation of the retrieval accuracy of the proposed approach. It showed that the approach achieved a good level of accuracy in classifying the document relevancy with a given user query term. It also showed a good retrieval performance. However the retrieval accuracy is expected to improve as the system continues to be used and as more data is captured from users. This chapter discusses the main contributions of the thesis and limitations of the research. It also discusses the fulfilment of the research objectives by relating them to the appropriate thesis chapters.

The rest of the chapter is organised as follows: Section 7.2 provides a summary of the research. Section 7.3 discusses the contribution of the research in relation to the research objectives. Section 7.4 discusses the research limitations. Section 7.5 discusses the future work.

7.2. RESEARCH SUMMARY

This research has presented an approach for the development of a fuzzy logic recommender system for enterprise search. The approach provides a new mechanism for constructing and integrating a task, user and document profile into a unified index, through the use of relevance feedback and fuzzy rule based summarisation. The fuzzy approach was used to create the profiles and integrate them. The motivation for using the fuzzy approach was to handle the uncertainty due to the inconsistency and subjectivity in the assessment of relevance feedback provided by the user.

The research included a series of experiments in which relevance feedback was captured from 35 users on 20 predefined simulated enterprise search tasks. During these experiments the system captured implicit and explicit feedback parameters, together with the user search queries. The captured relevance feedback was used to develop and train the fuzzy system. The system showed an 89% accuracy performance in correctly classifying document relevance. The overall retrieval accuracy of the recommender system was evaluated based on standard precision and recall which

showed significant improvements in retrieving relevant documents.

As discussed in the previous chapter, the proposed approach enhanced the values of both Recall and precision significantly which indicated that the number of the unwanted (i.e. irrelevant) documents was being minimised. This helped to overcome information overload by filtering out the unwanted information while maintaining a good retrieval accuracy performance. In addition the approach enhanced the ability of the system to show the relevant documents at the top of the search result page thus reducing the time consumed by the user to access the required information and in turn helped to address the problem of information overload. This tended to help the user spending the available time reading the relevant documents rather than wasting time searching for them.

7.3. CONTRIBUTION

The main contribution made in this thesis was the development of an adaptive integrated fuzzy approach for a recommender system to be used for the enterprise search. The recommender system was used to recommend relevant documents based on relevance feedback. In addition, the system also recommended people who have expertise in the search area. The proposed approach was adaptive because the rules and models which were used to give weight to the “expertise” of the people and the relevance of the documents were extracted from the relevance feedback data set. This helped both the model and corresponding rules to change dynamically based on the underlying data set. The proposed approach integrated both the implicit and explicit relevance feedback in order to estimate the relevance of the document to the user query. The relevance feedback (implicit and explicit) and fuzzy logic inference was used to create and integrate the task, user and document profiles. The fuzzy logic resolved the uncertainty and bias which were found in the user behaviour. Briefly, the following contributions were made to the existing knowledge:

- In order to explore the existing approaches used for enterprise search and to further understand the problem domain an extensive literature review was carried out and presented in *Chapter 2* and *Chapter 3*. The literature also covered the recommender system approaches, focusing on those approaches which were proposed for enterprise search. In addition the literature surveyed fuzzy based approaches for recommender systems.
- In order to investigate the relevance feedback including the relationship between

the implicit and explicit parameters, an extensive literature review was presented in **Chapter 3** and empirical research was presented in **Chapters 4, 5** and **5**. The empirical research carried out as part of this thesis clearly found significant co-relation between the implicit and explicit relevance feedback. Co-relation and regression analysis were used to analyse the relationships between implicit and explicit feedback parameters. As a result of this analysis, it was found that there was a linear relationship between the implicit parameters (i.e. time on page, mouse movements, and mouse clicks) and the explicit document relevance. The linear relationship was then translated into an adaptive linear predictive model to estimate the document relevance from the implicit feedback parameters. While the explicit feedback parameters were used to train and validate the approach, the accuracy of the model was evaluated using R-squared (R^2) and the model achieved 76% accuracy in predicting the visited document relevancy from the implicit parameters.

- To fill in the gaps in knowledge and to extend the existing body of knowledge a fuzzy based approach was proposed and carefully applied to an application area of enterprise search as discussed in **Chapter 5**. The proposed approach included two new mechanisms:

- A new approach for profiling was proposed. The proposed approach included a task, user and document profile. The intuition behind the building of three profiles was that, in the task based search the user searched for *information* which existed in the *documents* in order to complete a specific *task*. This meant that the user judgment of relevancy in the search task was dependent on all three of these. The proposed profiling approach was inspired by (Li & Kim, 2004). The approach was a fuzzy logic based method in which, first, a subset of the user queries was identified and selected based on the type of the profile being created. Then, the selected queries were pre-processed to extract the query terms using Porter's analysis and the DTF, DF, and IDF was calculated for each of each term. Then, the values of these matrices were fed into a fuzzy logic system to weight the relevance of the term to the search task, the user or the document.
- An adaptive fuzzy mechanism was developed to integrate the three profiles into one index that contained a unified term weight for each occurrence of the term in the user queries. The new term weight reflected the relevance of the term to the task, user and document which corresponded to the occurrence

instance. The three profiles were projected based on the query term. For each document visit instance, each query term was associated with its weight from the corresponding search task, user and document profiles and the predicted relevance level which was calculated by the linear predictive model. The associated values were given ‘High’, ‘Medium’ and ‘Low’ linguistic labels. These were represented in the “if→then” fuzzy rule form. Generality and consistency measures were used to summarize the fuzzy rules. This was done by calculating the support and confidence for each rule pattern. And then, the pattern with the higher firing strength was selected out of any conflicting patterns (patterns with same predecessor successor). Using the summarized fuzzy rules and the same linguistic labels the fuzzy inference system calculated a unified fuzzy weight for each term in each visit instance. The unified weight considered the relevance between the query term and the search task, the user and the document at same time.

- As a result of the research experiments, as discussed in **Chapter 4**, the labelled data of the well-known enterprise search test collection ‘TREC Enterprise Track 2007’ was extended to include more user queries for the topics provided and relevance feedback (implicit and explicit) on the created queries. A set of 20 simulated search tasks were designed based on the labelled data. The designed search tasks were given to 35 users to complete. The users were asked to create their own queries (Keywords) using the configured search system to complete the provided tasks. During this process, the system observed their search behaviour and captured their relevance feedback. The system captured 812 user queries and 1230 document visits which gave a reasonable amount of relevance feedback for creating the user, document and search task profiles.
- Furthermore, as discussed in **Chapter 5**, the proposed approach was implemented in a form of a prototype recommender system that recommended both people with the relevant expertise as well as the relevant documents for a specific user query. The system could be integrated with any other search system within the enterprise.
- In order to evaluate the retrieval accuracy of the proposed system, as discussed in **Chapter 6**, different measures and methods were used in order to ensure that all aspects of the proposed approach were tested thoroughly. Both the accuracy of the

component validation and the overall retrieval accuracy was evaluated. The accuracy performance of the linear predictive model was calculated using the R-squared method and the model showed 76.5 % accuracy in predicting the document relevance. A 5-fold cross validation method was applied to train and validate the fuzzy classifier, which was built based on the summarised fuzzy rules. The highest accuracy performance that the classifier showed in correctly classifying documents-terms relevance was 89% at the fold-4 and the average accuracy for all folds was 86.2% which was relatively good as an initial performance and it was expected to improve as more data was captured from more users. The overall retrieval accuracy of the proposed system was tested by carrying out a comparative retrieval accuracy evaluation based on Precision (**P**), Recall (**R**) and ranking analysis. The values of **P** and **R** of the proposed system were compared to both the Standard inverted index based Solr system and the semantic indexing based Lucid system. The proposed system enhanced the value of **P** significantly where the average of **P** value increased from 0.00428 to 0.064 compared with standard Solr system and from 0.0298 to 0.064 compared with Lucid system. In other words, the proposed approach managed to decrease the number of irrelevant documents in the search result which meant that the ability of the system to only show the relevant document was enhanced. The proposed system enhanced the value of **R** as well. The average value of **R** was increased significantly (doubling) from 0.436 to 0.828 compared with standard Solr system and from 0.76804 to 0.828 compared with Lucid system. This meant the ability of the system to retrieve the relevant document was also enhanced. Furthermore the ability of the system to rank more highly the relevant documents improved as compared with other two systems.

7.4. RESEARCH LIMITATIONS

As is the case with any research, readers need to consider the presented results within the context of the limitations. Also, the process of posing and answering particular research questions typically generates more questions that need to be explored through further research. With regard to current research the research results were limited to following:

- Lack of access to real world organization data and users. In general, organisations do not want their competitors to know what their employees might be searching for, or even what sets of documents they might be searching. For this

reason, query logs are unlikely to be made available for publication or even for perusal by external researchers (Hawking 2010). With regard to the current research, it was difficult to have the access to the internal data of any organisations due to the privacy and security concerns.

- The test collection (the data set), defining an appropriate enterprise search test collection for the experiments is not a straightforward task. The development of an enterprise search test collection which is sophisticated enough to model an interesting research problem and to serve as a benchmark by which the developed approaches could be tuned and improved, or, by which products may be compared, are still challenging tasks in the enterprise search research area (Hawking 2010). An alternative to developing the test collection is to use one of the standard test collections which are available for experimental purposes. The problem with such test collections, is that they have, to different extents, a lack of complete real world representation. For example, the test collection ‘TREC Enterprise 2007 Track’ which is a common test collection in the research community and was used in the current research, provides a relatively limited amount of labelled data. The labelled data is limited to 50 user queries, together with their associated user judgment. This number of queries might be adequate for evaluation purposes, but is not adequate for profiling or to develop any relevance feedback based approach.
- Considering the limited labelled data, a set of simulated search tasks were developed in order to extend the labelled data to model the real world situation more realistically: In order to extend the labelled data set, the developed search tasks were given to a group of 35 users to complete using the provided system and their own queries (keywords). But the challenge was how to make the participants who were not experienced in the test collection to behave in a way which reflected the real world situation. Due the time constraints and participants availability the only available way was to explain the search tasks thoroughly to them and also to train them on how to complete 5 trial search tasks.

7.5. FUTURE WORK

Future work will include further evaluation of the proposed approach in a real world organisation, using an extended user base and over a longer timescale. Some additional moderating factors such as the age of the query and document authorship will be investigated and included in the proposed approach. Furthermore, the

investigated implicit feedback parameters set, will be expanded to include parameters related to text section (e.g. highlighting and copying); creating and creating and updating parameters). The general linear predictive model will be replaced with a user specific predictive approach in which each user has their own model to predict the relevance of the document from the implicit feedback parameters based on their behavioural patterns. In the large organisations, information could be distributed on different geographical locations and also on different platforms (Hawking 2010). This research was mainly concerned with retrieval accuracy and therefore, other performance aspects such as the scalability of the proposed approach will be explored in future. Future work will consider scalability in order to produce a highly scalable recommender system and will also deploy the system on cloud based architecture in order to address the issues related to computation cost and distribution of data (Anjum et al. 2012).

8. REFERENCES

- Abel, F., Gao, Q., Houben, G., and Tao, K. (eds.) (2013) *Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence*. 'Twitter-Based User Modeling for News Recommendations': AAAI Press
- Afzal, U. and Islam, M. (2013) 'Meven: An Enterprise Trust Recommender System'
- Agichtein, E., Brill, E., Dumais, S., and Ragno, R. (eds.) (2006) *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 'Learning User Interaction Models for Predicting Web Search Result Preferences': ACM
- Ahmed, A., Das, A., and Smola, A. J. (eds.) (2014) *Proceedings of the 7th ACM International Conference on Web Search and Data Mining*. 'Scalable Hierarchical Multitask Learning Algorithms for Conversion Optimization in Display Advertising': ACM
- Alhabashneh, O., Iqbal, R., Shah, N., Amin, S., and James, A. (2011) 'Towards the Development of an Integrated Framework for Enhancing Enterprise Search using Latent Semantic Indexing'. in *Conceptual Structures for Discovering Knowledge*. ed. by Anon: Springer, 346-352
- Alpaydin, E. (2014) *Introduction to Machine Learning*.: MIT press
- Amatriain, X., Pujol, J. M., and Oliver, N. (2009) 'I Like it... i Like it Not: Evaluating User Ratings Noise in Recommender Systems'. in *User Modeling, Adaptation, and Personalization*. ed. by Anon: Springer, 247-258
- Amatriain, X., Pujol, J. M., Tintarev, N., and Oliver, N. (eds.) (2009) *Proceedings of the Third ACM Conference on Recommender Systems*. 'Rate it again: Increasing Recommendation Accuracy by User Re-Rating': ACM
- Anand, S. S., Kearney, P., and Shapcott, M. (2007) 'Generating Semantically Enriched User Profiles for Web Personalization'. *ACM Transactions on Internet Technology (TOIT)* 7 (4), 22
- Anjum, A., Hill, R., McClatchey, R., Bessis, N., and Branson, A. (2012) 'Glueing Grids and Clouds Together: A service-oriented Approach'. *International Journal of Web and Grid Services* 8 (3), 248-265
- Ardissono, L., Goy, A., Petrone, G., Segnan, M., and Torasso, P. (eds.) (2002) *Adaptive Hypermedia and Adaptive Web-Based Systems*. 'Ubiquitous User Assistance in a Tourist Information Server': Springer
- Arlot, S. and Celisse, A. (2010) 'A Survey of Cross-Validation Procedures for Model Selection'. *Statistics Surveys* 4, 40-79

- Baeza, R. and Ribeiro-Neto, B. (2011) 'Enterprise Search'. in *Modern Information Retrieval 2nd Ed.* ed. by Anon: Pearson Educational, 641-44
- Bailey, P., De Vries, A. P., Craswell, N., and Soboroff, I. (eds.) (2007) *Trec. 'Overview of the TREC 2007 Enterprise Track.'*: Citeseer
- Baker, C., Anjum, A., Hill, R., Bessis, N., and Kiani, S. L. (eds.) (2012) *Intelligent Networking and Collaborative Systems (INCoS), 2012 4th International Conference on.* 'Improving Cloud Datacentre Scalability, Agility and Performance using OpenFlow': IEEE
- Balakrishnan, V. and Zhang, X. (2014) 'Implicit User Behaviours to Improve Post-Retrieval Document Relevancy'. *Computers in Human Behavior* 33, 104-112
- Balog, K., Azzopardi, L., and de Rijke, M. (2009) 'A Language Modeling Framework for Expert Finding'. *Information Processing & Management* 45 (1), 1-19
- Balog, K. and De Rijke, M. (2008) . *Combining Candidate and Document Models for Expert Search*
- Balog, K., Bogers, T., Azzopardi, L., De Rijke, M., and Van Den Bosch, A. (eds.) (2007) *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* 'Broad Expertise Retrieval in Sparse Data Environments': ACM
- Balog, K., Azzopardi, L., and De Rijke, M. (eds.) (2006) *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval.* 'Formal Models for Expert Finding in Enterprise Corpora': ACM
- Banwell, L. and Coulson, G. (2004) 'Users and User Study Methodology: The JUBILEE Project'. *Information Research* 9 (2), 9-2
- Bao, Z., Kimelfeld, B., and Li, Y. (eds.) (2012) *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval.* 'Automatic Suggestion of Query-Rewrite Rules for Enterprise Search': ACM
- Bell, R. M. and Koren, Y. (eds.) (2007) *KDD Cup and Workshop at the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining.* 'Improved Neighborhood-Based Collaborative Filtering': sn
- Bidoki, A. M. Z., Ghodsnia, P., Yazdani, N., and Oroumchian, F. (2010) 'A3CRank: An Adaptive Ranking Method Based on Connectivity, Content and Click-through Data'. *Information Processing & Management* 46 (2), 159-169
- Bobadilla, J., Ortega, F., Hernando, A., and Gutiérrez, A. (2013) 'Recommender Systems Survey'. *Knowledge-Based Systems* 46, 109-132

- Bostandjiev, S., O'Donovan, J., and Höllerer, T. (eds.) (2012) *Proceedings of the Sixth ACM Conference on Recommender Systems*. 'Tasteweights: A Visual Interactive Hybrid Recommender System': ACM
- Bouza, A., Reif, G., Bernstein, A., and Gall, H. (eds.) (2008) *International Semantic Web Conference (Posters & Demos)*. 'SemTree: Ontology-Based Decision Tree Algorithm for Recommender Systems.': Citeseer
- Breese, J. S., Heckerman, D., and Kadie, C. (eds.) (1998) *Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence*. 'Empirical Analysis of Predictive Algorithms for Collaborative Filtering': Morgan Kaufmann Publishers Inc.
- Broder, A. Z. and Ciccolo, A. C. (2004) 'Towards the Next Generation of Enterprise Search Technology'. *IBM Systems Journal* 43 (3), 451-454
- Budzik, J. and Hammond, K. (eds.) (1999) *Proceedings of the Annual Meeting-American Society for Information Science*. 'Watson: Anticipating and Contextualizing Information Needs': Citeseer
- Burke, R. (2007) 'Hybrid Web Recommender Systems'. in *The Adaptive Web*. ed. by Anon: Springer, 377-408
- Buscher, G., White, R. W., Dumais, S., and Huang, J. (eds.) (2012) *Proceedings of the Fifth ACM International Conference on Web Search and Data Mining*. 'Large-Scale Analysis of Individual and Task Differences in Search Result Page Examination Strategies': ACM
- Cao, Y., Liu, J., Bao, S., and Li, H. (eds.) (2005) *Trec*. 'Research on Expert Search at Enterprise Track of TREC 2005.'
- Carbo, J. and Molina, J. M. (2004) 'Agent-Based Collaborative Filtering Based on Fuzzy Recommendations'. *International Journal of Web Engineering and Technology* 1 (4), 414-426
- Castellanos, M. (2004) 'Hotminer: Discovering Hot Topics from Dirty Text'. in *Survey of Text Mining*. ed. by Anon: Springer, 123-157
- Christakou, C., Vrettos, S., and Stafylopatis, A. (2007) 'A Hybrid Movie Recommender System Based on Neural Networks'. *International Journal on Artificial Intelligence Tools* 16 (05), 771-792
- Cintra, M. E., Monard, M. C., Camargo, H. A., and Martin, T. P. (2005) 'A Comparative Study on Classic Machine Learning and Fuzzy Approaches for Classification Problems.'"
- Claypool, M., Le, P., Wased, M., and Brown, D. (eds.) (2001) *Proceedings of the 6th International Conference on Intelligent User Interfaces*. 'Implicit Interest Indicators': ACM

- Collins-Thompson, K., Bennett, P. N., White, R. W., de la Chica, S., and Sontag, D. (eds.) (2011) *Proceedings of the 20th ACM International Conference on Information and Knowledge Management*. 'Personalizing Web Search Results by Reading Level': ACM
- Colomo-Palacios, R., González-Carrasco, I., López-Cuadrado, J. L., and García-Crespo, Á. (2012) 'ReSySTER: A Hybrid Recommender System for Scrum Team Roles Based on Fuzzy and Rough Sets'. *International Journal of Applied Mathematics and Computer Science* 22 (4), 801-816
- Cornelis, C., Lu, J., Guo, X., and Zhang, G. (2007) 'One-and-Only Item Recommendation with Fuzzy Logic Techniques'. *Information Sciences* 177 (22), 4906-4921
- Craswell, N., Hawking, D., Vercoustre, A., and Wilkins, P. (eds.) (2001) *Ausweb Poster Proceedings, Queensland, Australia*. 'P@ Noptic Expert: Searching for Experts Not just for Documents'
- Dan, C. and Sherlock, C. (2008) *Introduction to Regression and Data Analysis* , October 28, . Workshop Series edn: Statlab
- Das, A. S., Datar, M., Garg, A., and Rajaram, S. (eds.) (2007) *Proceedings of the 16th International Conference on World Wide Web*. 'Google News Personalization: Scalable Online Collaborative Filtering': ACM
- de Campos, L. M., Fernández-Luna, J. M., and Huete, J. F. (2008) 'A Collaborative Recommender System Based on Probabilistic Inference from Fuzzy Observations'. *Fuzzy Sets and Systems* 159 (12), 1554-1576
- Delic, K. A. and Riley, J. A. (eds.) (2009) *Eknow*. 'Enterprise Knowledge Clouds: Next Generation KM Systems?.'
- Deng, H., Han, J., Lyu, M. R., and King, I. (eds.) (2012) *Proceedings of the 12th ACM/IEEE-CS Joint Conference on Digital Libraries*. 'Modeling and Exploiting Heterogeneous Bibliographic Networks for Expertise Ranking': ACM
- Deng, H., King, I., and Lyu, M. R. (eds.) (2008) *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*. 'Formal Models for Expert Finding on Dblp Bibliography Data': IEEE
- Dmitriev, P., Serdyukov, P., and Chernov, S. (eds.) (2010) *Proceedings of the 19th International Conference on World Wide Web*. 'Enterprise and Desktop Search': ACM
- Dmitriev, P. A., Eiron, N., Fontoura, M., and Shekita, E. (eds.) (2006) *Proceedings of the 15th International Conference on World Wide Web*. 'Using Annotations in Enterprise Search': ACM

- Doctor, F. and Iqbal, R. (eds.) (2012) *Fuzzy Systems (FUZZ-IEEE), 2012 IEEE International Conference on*. 'An Intelligent Framework for Monitoring Student Performance using Fuzzy Rule-Based Linguistic Summarisation': IEEE
- Doctor, F., Hagra, H., Roberts, D., and Callaghan, V. (eds.) (2009) *Intelligent Agents, 2009. IA'09. IEEE Symposium on*. 'A Fuzzy Based Agent for Group Decision Support of Applicants Ranking within Recruitment Systems': IEEE
- Durand, G., Laplante, F., and Kop, R. (eds.) (2011) *KDD-2011: 17th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 'A Learning Design Recommendation System Based on Markov Decision Processes'
- Eckhardt, A. (2012) 'Similarity of Users'(Content-Based) Preference Models for Collaborative Filtering in Few Ratings Scenario'. *Expert Systems with Applications* 39 (14), 11511-11516
- Elkahky, A., Song, Y., and He, X. 'A Multi-View Deep Learning Approach for Cross Domain User Modeling in Recommendation Systems'
- Fagin, R., Kumar, R., McCurley, K. S., Novak, J., Sivakumar, D., Tomlin, J. A., and Williamson, D. P. (eds.) (2003) *Proceedings of the 12th International Conference on World Wide Web*. 'Searching the Workplace Web': ACM
- Fang, H. and Zhai, C. (2007) *Probabilistic Models for Expert Finding*.: Springer
- Feldman, S. (2004) *The High Cost of Not Finding Information*.: Information Today, Incorporated
- FindWise, (2015) *FindWise Survey on Information Findability 2013* [online] available from <<http://www.slideshare.net/findwise/enterprise-search-and-findability-in-2013>> [2015]
- Fox, S., Karnawat, K., Mydland, M., Dumais, S., and White, T. (2005) 'Evaluating Implicit Measures to Improve Web Search'. *ACM Transactions on Information Systems (TOIS)* 23 (2), 147-168
- Freund, L. and Toms, E. G. (eds.) (2006) *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 'Enterprise Search Behaviour of Software Engineers': ACM
- Freund, L., Toms, E. G., and Clarke, C. L. (eds.) (2005) *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 'Modeling Task-Genre Relationships for IR in the Workplace': ACM
- Gao, H., Tang, J., Hu, X., and Liu, H. (eds.) (2015) . 'Content-Aware Point of Interest Recommendation on Location-Based Social Networks': AAAI
- Gao, Y., Ilves, K., and Głowacka, D. (eds.) (2015) *Proceedings of the 20th International Conference on Intelligent User Interfaces Companion*. 'OfficeHours:

- A System for Student Supervisor Matching through Reinforcement Learning': ACM
- Garcia-Perez, A., Shaikh, S. A., Kalutarage, H. K., Jahantab, M., and Chase, R. (2015) 'Towards a Knowledge-Based Approach for Effective Decision Making in Railway Safety'. *Journal of Knowledge Management* 19 (3)
- Gershman, A., Meisels, A., Lüke, K., Rokach, L., Schclar, A., and Sturm, A. (eds.) (2010) *Iics*. 'A Decision Tree Based Recommender System.': Citeseer
- Ghazanfar, M. and Prugel-Bennett, A. (2010) 'An Improved Switching Hybrid Recommender System using Naive Bayes Classifier and Collaborative Filtering'
- Gollapalli, S. D., Mitra, P., and Giles, C. L. (eds.) (2012) *Proceedings of the 12th ACM/IEEE-CS Joint Conference on Digital Libraries*. 'Similar Researcher Search in Academic Environments': ACM
- Golovchinsky, G., Price, M. N., and Schilit, B. N. (eds.) (1999) *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 'From Reading to Retrieval: Freeform Ink Annotations as Queries': ACM
- Grzywaczewski, A. and Iqbal, R. (2012) 'Task-Specific Information Retrieval Systems for Software Engineers'. *Journal of Computer and System Sciences* 78 (4), 1204-1218
- Gulliksen, J., Göransson, B., Boivie, I., Blomkvist, S., Persson, J., and Cajander, Å. (2003) 'Key Principles for User-Centred Systems Design'. *Behaviour and Information Technology* 22 (6), 397-409
- Gunawardana, A. and Meek, C. (eds.) (2008) *Proceedings of the 2008 ACM Conference on Recommender Systems*. 'Tied Boltzmann Machines for Cold Start Recommendations': ACM
- Guo, Q. and Agichtein, E. (eds.) (2012) *Proceedings of the 21st International Conference on World Wide Web*. 'Beyond Dwell Time: Estimating Document Relevance from Cursor Movements and Other Post-Click Searcher Behavior': ACM
- Hansen, P. and Järvelin, K. (eds.) (2000) *Proc. ACM SIGIR Workshop on Patent Retrieval*. 'The Information Seeking and Retrieval Process at the Swedish Patent and Registration Office'
- Hawking, D. (2010) 'Enterprise Search'. *Modern Information Retrieval*, 641-683
- Hawking, D. and Zobel, J. (2007) 'Does Topic Metadata Help with Web Search?'. *Journal of the American Society for Information Science and Technology* 58 (5), 613-628

- Hawking, D., Paris, C., Wilkinson, R., and Wu, M. (eds.) (2005) *Proc. IRiX Workshop, ACM SIGIR*. 'Context in Enterprise Search and Delivery'
- Hawking, D. (ed.) (2004) *Proceedings of the 15th Australasian Database Conference-Volume 27*. 'Challenges in Enterprise Search': Australian Computer Society, Inc.
- Hawking, D., Crimmins, F., Craswell, N., and Upstill, T. (eds.) (2004) *Proceedings of the 15th Australasian Database Conference-Volume 27*. 'How Valuable is External Link Evidence when Searching Enterprise Webs?': Australian Computer Society, Inc.
- Hertzum, M. and Pejtersen, A. M. (2000) 'The Information-Seeking Practices of Engineers: Searching for Documents as Well as for People'. *Information Processing & Management* 36 (5), 761-778
- Hofmann, T. (ed.) (2003) *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*. 'Collaborative Filtering Via Gaussian Probabilistic Latent Semantic Analysis': ACM
- Holz, H., Rostanin, O., Dengel, A., Suzuki, T., Maeda, K., and Kanasaki, K. (eds.) (2006) *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*. 'Task-Based Process Know-how Reuse and Proactive Information Delivery in TaskNavigator': ACM
- Hu, Y., Koren, Y., and Volinsky, C. (eds.) (2008) *Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on*. 'Collaborative Filtering for Implicit Feedback Datasets': IEEE
- IDC , (2005) *The Enterprise Workplace: How it Will Change the Way we Work* : IDC Report 32919
- Idinopulos, M. and Kempler, L. (2006) 'Do You Know Who Your Experts are?'
- Iqbal, R., Doctor, F., Shah, N., and Fei, X. (2014) 'An Intelligent Framework for Activity Led Learning in Network Planning and Management'. *International Journal of Communication Networks and Distributed Systems* 12 (4), 401-419
- Ishibuchi, H. and Yamamoto, T. (2005) 'Rule Weight Specification in Fuzzy Rule-Based Classification Systems'. *Fuzzy Systems, IEEE Transactions on* 13 (4), 428-435
- Jackson, W. K. (2001) *Information Overload and Managerial Roles: A Naturalistic Study of Engineers*.
- Joachims, T., Granka, L., Pan, B., Hembrooke, H., Radlinski, F., and Gay, G. (2007) 'Evaluating the Accuracy of Implicit Feedback from Clicks and Query Reformulations in Web Search'. *ACM Transactions on Information Systems (TOIS)* 25 (2), 7

- Jung, S., Herlocker, J. L., and Webster, J. (2007) 'Click Data as Implicit Relevance Feedback in Web Search'. *Information Processing & Management* 43 (3), 791-807
- Kantor, P. B., Rokach, L., Ricci, F., and Shapira, B. (2011) *Recommender Systems Handbook*.: Springer
- Kelly, D. (2009) 'Methods for Evaluating Interactive Information Retrieval Systems with Users'. *Foundations and Trends in Information Retrieval* 3 (1—2), 1-224
- Kelly, D. and Teevan, J. (eds.) (2003) *ACM SIGIR Forum*. 'Implicit Feedback for Inferring User Preference: A Bibliography': ACM
- Kleinberg, J. M. (1999) 'Authoritative Sources in a Hyperlinked Environment'. *Journal of the ACM (JACM)* 46 (5), 604-632
- Lee, J., Bengio, S., Kim, S., Lebanon, G., and Singer, Y. (eds.) (2014) *Proceedings of the 23rd International Conference on World Wide Web*. 'Local Collaborative Ranking': International World Wide Web Conferences Steering Committee
- Lee, M., Choi, P., and Woo, Y. (eds.) (2002) *Adaptive Hypermedia and Adaptive Web-Based Systems*. 'A Hybrid Recommender System Combining Collaborative Filtering with Neural Network': Springer
- Li, B., Yang, Q., and Xue, X. (eds.) (2009) *Proceedings of the 26th Annual International Conference on Machine Learning*. 'Transfer Learning for Collaborative Filtering Via a Rating-Matrix Generative Model': ACM
- Li, P. and Yamada, S. (eds.) (2004) *Cybernetics and Intelligent Systems, 2004 IEEE Conference on*. 'A Movie Recommender System Based on Inductive Learning': IEEE
- Li, Q. and Kim, B. M. (2004) 'Constructing User Profiles for Collaborative Recommender System'. in *Advanced Web Technologies and Applications*. ed. by Anon: Springer, 100-110
- Linden, G., Smith, B., and York, J. (2003) 'Amazon. Com Recommendations: Item-to-Item Collaborative Filtering'. *Internet Computing, IEEE* 7 (1), 76-80
- Liu, J., Dolan, P., and Pedersen, E. R. (eds.) (2010) *Proceedings of the 15th International Conference on Intelligent User Interfaces*. 'Personalized News Recommendation Based on Click Behavior': ACM
- Liu, X., Fang, H., Chen, F., and Wang, M. (eds.) (2012) *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*. 'Entity Centric Query Expansion for Enterprise Search': ACM
- Liu, Z. and Sun, M. (eds.) (2008) *Proceedings of the 17th International Conference on World Wide Web*. 'Asymmetrical Query Recommendation Method Based on Bipartite Network Resource Allocation': ACM

- Lü, L., Medo, M., Yeung, C. H., Zhang, Y., Zhang, Z., and Zhou, T. (2012) 'Recommender Systems'. *Physics Reports* 519 (1), 1-49
- LucidWorks, (2015) [online] available from <<https://lucidworks.com/resources/>> [2015]
- Macdonald, C. and Ounis, I. (2008) 'Voting Techniques for Expert Search'. *Knowledge and Information Systems* 16 (3), 259-280
- Magnusson, C., Rasmussen-Gröhn, K., Tollmar, K., and Deaner, E. (2009) *User Study Guidelines: Hapti Map Consorti um*
- Mahmood, T. and Ricci, F. (eds.) (2009) *Proceedings of the 20th ACM Conference on Hypertext and Hypermedia*. 'Improving Recommender Systems with Adaptive Conversational Strategies': ACM
- Mamdani, E. (1974) 'Applications of Fuzzy Logic for Control of a Simple Dynamic Process'. *Proc IEEE* 121, 1585-1588
- Mandel, J. (2012) *The Statistical Analysis of Experimental Data.*: Courier Corporation
- Mangold, C., Schwarz, H., and Mitschang, B. (eds.) (2006) *Database Engineering and Applications Symposium, 2006. IDEAS'06. 10th International*. 'U38: A Framework for Database-Supported Enterprise Document-Retrieval': IEEE
- Manuja, K. and Bhattacharya, A. (eds.) (2015) *Proceedings of the Second ACM IKDD Conference on Data Sciences*. 'Using Social Connections to Improve Collaborative Filtering': ACM
- Masthoff, J. (2004) 'Group Modeling: Selecting a Sequence of Television Items to Suit a Group of Viewers'. in *Personalized Digital Television*. ed. by Anon: Springer, 93-141
- Maybury, M. T. (2006) 'Expert Finding Systems'. *MITRE Center for Integrated Intelligence Systems Bedford, Massachusetts, USA*
- McCandless, M., Hatcher, E., and Gospodnetic, O. (2010) *Lucene in Action: Covers Apache Lucene 3.0.*: Manning Publications Co.
- McCarthy, K., Salamó, M., Coyle, L., McGinty, L., Smyth, B., and Nixon, P. (eds.) (2006) *Proceedings of the 11th International Conference on Intelligent User Interfaces*. 'Group Recommender Systems: A Critiquing Based Approach': ACM
- Melville, P., Mooney, R. J., and Nagarajan, R. (eds.) (2002) *Aaai/iaai*. 'Content-Boosted Collaborative Filtering for Improved Recommendations'
- Morita, M. and Shinoda, Y. (eds.) (1994) *Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 'Information Filtering Based on User Behavior Analysis and Best Match Text Retrieval': Springer-Verlag New York, Inc.

- Mukherjee, R. and Mao, J. (2004) 'Enterprise Search: Tough Stuff'. *Queue* 2 (2), 36
- Namahoot, C. S., Brückner, M., and Panawong, N. (2015) 'Context-Aware Tourism Recommender System using Temporal Ontology and Naïve Bayes'. in *Recent Advances in Information and Communication Technology 2015*. ed. by Anon: Springer, 183-194
- Oard, D. W. and Kim, J. (2001) 'Modeling Information Content using Observable Behavior'
- Owens, L., Brown, M., Poore, K., and Nicolson, N. (2008) 'The Forrester Wave: Enterprise Search, Q2 2008'. *For Information and Knowledge Management Professionals*
- Page, L., Brin, S., Motwani, R., and Winograd, T. (1999) 'The PageRank Citation Ranking: Bringing Order to the Web.'
- Pan, W., Xiang, E. W., Liu, N. N., and Yang, Q. (eds.) (2010) *Aaai*. 'Transfer Learning in Collaborative Filtering for Sparsity Reduction.'
- Patel, M. X., Doku, V., and Tennakoon, L. (2003) 'Challenges in Recruitment of Research Participants'. *Advances in Psychiatric Treatment* 9 (3), 229-238
- Perny, P. and Zucker, J. (2001) 'Preference-Based Search and Machine Learning for Collaborative Filtering: The 'Film-Conseil' Movie Recommender System'. *Revue I3* 1 (1), 1-40
- Poblete, B. and Baeza-Yates, R. (eds.) (2008) *Proceedings of the 17th International Conference on World Wide Web*. 'Query-Sets: Using Implicit Feedback and Query Patterns to Organize Web Documents': ACM
- Porter, M. F. (1980) 'An Algorithm for Suffix Stripping'. *Program* 14 (3), 130-137
- Rafter, R. and Smyth, B. (eds.) (2001) *Workshop on Intelligent Techniques for Web Personalization at the the 17th International Joint Conference on Artificial Intelligence*. 'Passive Profiling from Server Logs in an Online Recruitment Environment' at USA: University College Dublin
- Raghavan, P. (2002) 'Social Networks: From the Web to the Enterprise'. *Internet Computing, IEEE* 6 (1), 91-94
- Raghavan, P. (2001) 'Structured and Unstructured Search in Enterprises'. *IEEE Data Eng.Bull.* 24 (4), 15-18
- Ramachandran, P. (ed.) (2005) *Discovery Science*. 'Discovering User Preferences by using Time Entries in Click-through Data to Improve Search Engine Results': Springer
- Ren, L., He, L., Gu, J., Xia, W., and Wu, F. (eds.) (2008) *Future Generation Communication and Networking, 2008. FGCN'08. Second International*

- Conference on. 'A Hybrid Recommender Approach Based on Widrow-Hoff Learning': IEEE
- Rennie, J. D. and Srebro, N. (eds.) (2005) *Proceedings of the 22nd International Conference on Machine Learning*. 'Fast Maximum Margin Matrix Factorization for Collaborative Prediction': ACM
- Ricci, F., Rokach, L., Shapira, B., and Kantor, P. B. (2011) *Recommender Systems Handbook*.: Springer
- Rubin, J. and Chisnell, D. (2011) *Handbook of Usability Testing: How to Plan, Design, and Conduct Effective Tests*. 2nd edn: John Wiley & Sons
- Ruff, J. (2002) 'Information Overload: Causes, Symptoms and Solutions'. *Harvard Graduate School of Education*, 1-13
- Sahebi, S. and Brusilovsky, P. (2013) 'Cross-Domain Collaborative Recommendation in a Cold-Start Context: The Impact of User Profile Size on the Quality of Recommendation'. in *User Modeling, Adaptation, and Personalization*. ed. by Anon: Springer, 289-295
- Salakhutdinov, R. and Mnih, A. (eds.) (2008) *Proceedings of the 25th International Conference on Machine Learning*. 'Bayesian Probabilistic Matrix Factorization using Markov Chain Monte Carlo': ACM
- Salakhutdinov, R., Mnih, A., and Hinton, G. (eds.) (2007) *Proceedings of the 24th International Conference on Machine Learning*. 'Restricted Boltzmann Machines for Collaborative Filtering': ACM
- Sarwar, B., Karypis, G., Konstan, J., and Riedl, J. (eds.) (2001) *Proceedings of the 10th International Conference on World Wide Web*. 'Item-Based Collaborative Filtering Recommendation Algorithms': ACM
- Schafer, J. B., Frankowski, D., Herlocker, J., and Sen, S. (2007) 'Collaborative Filtering Recommender Systems'. in *The Adaptive Web*. ed. by Anon: Springer, 291-324
- Schein, A. I., Popescul, A., Ungar, L. H., and Pennock, D. M. (eds.) (2002) *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 'Methods and Metrics for Cold-Start Recommendations': ACM
- Schiaffino, S. and Amandi, A. (2009) 'Intelligent User Profiling'. in *Artificial Intelligence an International Perspective*. ed. by Anon: Springer, 193-216
- Seleng, M., Laclavik, M., Dlugolinsky, S., Ciglan, M., Tomasek, M., and Hluchy, L. (eds.) (2014) *Intelligent Engineering Systems (INES), 2014 18th International Conference on*. 'Approach for Enterprise Search and Interoperability using Lightweight Semantic': IEEE

- Serdyukov, P., Hiemstra, D., and Aly, R. (2009) 'Using the Global Web as an Expertise Evidence Source'
- Skalistis, S., Petrovic, D., and Shaikh, S. A. 'Operating Heavy Duty Vehicles Under Extreme Heat Conditions: A Fuzzy Approach for Smart Gear-Shifting Strategy'
- Skalistis, S., Petrovic, D., and Shaikh, S. A. (eds.) (2013) *Intelligent Transportation Systems - (ITSC), 2013 16th International IEEE Conference on*. 'Operating Heavy Duty Vehicles Under Extreme Heat Conditions: A Fuzzy Approach for Smart Gear-Shifting Strategy'
- Smiley, D. and Pugh, E. (2009) *Solr 1.4 Enterprise Search Server.*: Packt Publishing Ltd
- Smyth, B., Balfe, E., Boydell, O., Bradley, K., Briggs, P., Coyle, M., and Freyne, J. (eds.) (2005) *Ijcai*. 'A Live-User Evaluation of Collaborative Web Search'
- Smyth, B., Balfe, E., Freyne, J., Briggs, P., Coyle, M., and Boydell, O. (2004) 'Exploiting Query Repetition and Regularity in an Adaptive Community-Based Web Search Engine'. *User Modeling and User-Adapted Interaction* 14 (5), 383-423
- Song, Y., Cui, W., Liu, S., and Wang, K. (eds.) (2014) *Proceedings of the Companion Publication of the 23rd International Conference on World Wide Web Companion*. 'Online Behavioral Genome Sequencing from Usage Logs: Decoding the Search Behaviors': International World Wide Web Conferences Steering Committee
- Song, Y., Wang, H., and He, X. (eds.) (2014) *Proceedings of the 7th ACM International Conference on Web Search and Data Mining*. 'Adapting Deep Ranknet for Personalized Search': ACM
- Su, X. and Khoshgoftaar, T. M. (2009) 'A Survey of Collaborative Filtering Techniques'. *Advances in Artificial Intelligence* 2009, 4
- Sun, M., Li, F., Lee, J., Zhou, K., Lebanon, G., and Zha, H. (eds.) (2013) *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*. 'Learning Multiple-Question Decision Trees for Cold-Start Recommendation': ACM
- Suresh, J. and Mahesh, K. (2006) *Ten Steps to Maturity in Knowledge Management: Lessons in Economy.*: Elsevier
- Tyler, S. K. and Teevan, J. (eds.) (2010) *Proceedings of the Third ACM International Conference on Web Search and Data Mining*. 'Large Scale Query Log Analysis of Re-Finding': ACM
- Tyler, S. K., Wang, J., and Zhang, Y. (eds.) (2010) *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*. 'Utilizing Re-Finding for Personalized Information Retrieval': ACM

- Upstill, T., Craswell, N., and Hawking, D. (2003) 'Query-Independent Evidence in Home Page Finding'. *ACM Transactions on Information Systems (TOIS)* 21 (3), 286-313
- Van den Oord, A., Dieleman, S., and Schrauwen, B. (eds.) (2013) *Advances in Neural Information Processing Systems*. 'Deep Content-Based Music Recommendation'
- Venkateshprasanna, H., Gandhi, R. D., Mahesh, K., and Suresh, J. (eds.) (2011) *Proceedings of the Fourth Annual ACM Bangalore Conference*. 'Enterprise Search through Automatic Synthesis of Tag Clouds': ACM
- Vozalis, M. G. and Margaritis, K. G. (2007) 'Using SVD and Demographic Data for the Enhancement of Generalized Collaborative Filtering'. *Information Sciences* 177 (15), 3017-3037
- Wang, C. and Blei, D. M. (eds.) (2011) *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 'Collaborative Topic Modeling for Recommending Scientific Articles': ACM
- Wang, G. A., Jiao, J., Abrahams, A. S., Fan, W., and Zhang, Z. (2013) 'ExpertRank: A Topic-Aware Expert Finding Algorithm for Online Knowledge Communities'. *Decision Support Systems* 54 (3), 1442-1451
- WANG, W. and CHEN, L. (2014) 'A Class Similarity Based Weight Estimation Algorithm for the Personalized Enterprise Search Engines'. *Journal of Computational Information Systems* 10 (5), 1903-1910
- White, M., Nikolov, S. G., Monteleone, S., Compañó, R., and Maghiros, I. (2013) *Enterprise Search in the European Union: A Techno-Economic Analysis*.: Publications Office
- White, R. W. and Buscher, G. (eds.) (2012) *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 'Text Selections as Implicit Relevance Feedback': ACM
- White, R. W., Ruthven, I., and Jose, J. M. (eds.) (2002) *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. 'Finding Relevant Documents using Top Ranking Sentences: An Evaluation of Two Alternative Schemes': ACM
- White, R. W., Ruthven, I., and Jose, J. M. (2002) 'The use of Implicit Evidence for Relevance Feedback in Web Retrieval'. in *Advances in Information Retrieval*. ed. by Anon: Springer, 93-109
- Witten, I. H. and Frank, E. (2005) *Data Mining: Practical Machine Learning Tools and Techniques*.: Morgan Kaufmann
- Wu, D., Mendel, J. M., and Joo, J. (eds.) (2010) *Fuzzy Systems (FUZZ), 2010 IEEE International Conference on*. 'Linguistic Summarization using IF-THEN Rules': IEEE

- Wu, M., Thom, J. A., Turpin, A., and Wilkinson, R. (eds.) (2009) *Proceedings of the 9th ACM/IEEE-CS Joint Conference on Digital Libraries*. 'Cost and Benefit Analysis of Mediated Enterprise Search': ACM
- Xie, W., Dong, Q., and Gao, H. (2014) 'A Probabilistic Recommendation Method Inspired by Latent Dirichlet Allocation Model'. *Mathematical Problems in Engineering* 2014
- Yager, R. R. (2003) 'Fuzzy Logic Methods in Recommender Systems'. *Fuzzy Sets and Systems* 136 (2), 133-149
- Yu, Z., Zhou, X., Hao, Y., and Gu, J. (2006) 'TV Program Recommendation for Multiple Viewers Based on User Profile Merging'. *User Modeling and User-Adapted Interaction* 16 (1), 63-82
- Zadeh, L. (1965) 'Fuzzy Logic and its Applications'. *New York, NY, USA*
- Zadeh, L. A. (1984) 'Making Computers Think Like People: The Termfuzzy Thinking_z is Pejorative when Applied to Humans, but Fuzzy Logic is an Asset to Machines in Applications from Expert Systems to Process Control'. *Spectrum, IEEE* 21 (8), 26-32
- Zadeh, L. A. (1973) 'Outline of a New Approach to the Analysis of Complex Systems and Decision Processes'. *Systems, Man and Cybernetics, IEEE Transactions on* (1), 28-44
- Zbigniew, M. .: - Springer Berlin Heidelberg
- Zhen, L., Huang, G. Q., and Jiang, Z. (2010) 'An Inner-Enterprise Knowledge Recommender System'. *Expert Systems with Applications* 37 (2), 1703-1712
- Zhen, L., Huang, G. Q., and Jiang, Z. (2009) 'Recommender System Based on Workflow'. *Decision Support Systems* 48 (1), 237-245
- Zhou, D., Orshanskiy, S. A., Zha, H., and Giles, C. L. (eds.) (2007) *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*. 'Co-Ranking Authors and Documents in a Heterogeneous Network': IEEE
- Zhou, Z. (2004) 'Rule Extraction: Using Neural Networks Or for Neural Networks?'. *Journal of Computer Science and Technology* 19 (2), 249-253

9. APPENDICES

9.1. APPENDEX1: INFORMATION SHEET & SEARCH TASK

INFORMATION SHEET

Title of Project: An Adaptive Fuzzy Logic Based Recommender System for Enterprise Search

Name of Researcher: OBADA. Y. ALHABASHNEH

You are being invited to take part in a research study. Before you decide it is important for you to understand why the research is being done and what it will involve. Please take the time to read the following information carefully. Ask me if there is anything that is not clear or if you would like more information.

The aim of this experiment is to enhance the enterprise search by using user feedback. We cannot capture and use the user feedback unless we ask real users to use our enterprise search engine, which is why we need to run experiments like these. Please remember that it is the search engine, not you, that is being evaluated and enhanced.

It is up to you to decide whether or not to take part. If you decide to take part you will be given this information sheet to keep and asked to sign a consent form. If you decide not to take part you are free to withdraw at any time without giving a reason. You also have the right to withdraw retrospectively any consent given, and to require that any data gathered on you be destroyed. A decision not to participate will not affect your grades in any way.

The experiment consists of 20 search tasks that you will be asked to carry out in order to retrieve the required information for a group of simulated work situations. During performing the search tasks the system will capture some implicit feedback indicators (such as reading time, scrolling, number of mouse clicks and others) and

explicit feedback (you will be asked to give a rate between 1 and 10 of how relevant the document is to your query). You will be given two weeks to finish the required tasks and each time you use the search system you will be asked to login.

Data will be stored only for analysis, and then will be destroyed. No personal data will be recorded. The results of this study will be used for my Ph.D research. The results are likely to be published in late 2014. You can request a summary of the results in the consent form. You will not be identified in any report or publication that arises from this work.

For further information about this experiment please contact:

Obada Alhabashneh (Email: aa9048@Coventry.ac.uk or Mobile: 07450490899).
Computing Department, Coventry University
Priory Street
Coventry, United Kingdom
CV1 5FB

Introduction

“CSIRO, the Commonwealth Scientific and Industrial Research Organisation, is the Australia's national science agency and one of the largest and most diverse research agencies in the world. It is a geographically distributed organisation at over 50 sites in Australia and around the world. It has 17 divisions, conducting research in areas such as entomology, industrial physics, mining, livestock, sustainable ecosystems and information & communication technologies.

One of the important roles in CSIRO is the science communicator's role which is to enhance CSIRO's public image and promote the capabilities of CSIRO by managing information and interacting with industry groups, government agencies, professional groups, media and the general public. Therefore, the communicators are supposed to have sufficient knowledge of various research activities taking place in the organisation.

In order to keep you up to date with the knowledge and organisational activities, as a communicator, you are asked to search information using your own search criteria on specific topics (see tasks list below) and familiarise yourself with the information. The information should be searched and retrieved using the provided search engine which is available on the following link: (<http://54.84.91.70>).

1. To find information on Gene Technology /Biotechnology focusing on Genetically Modified (GM) products such as cotton. In addition, find out how RNAi Technology is linked to the Gene Technology.
2. To find information about the likely impact of global climate change in Australia (including the social and environmental impacts).
3. To find at least 5 publications which you would recommend to get information about bushfires (forest fire) and fire in general that helps to develop better technologies and strategies to save lives. Those publications might also be recommended for protecting your home from bushfires and

improving house design to mitigate fires, etc.

4. To find information about safety and precautions for the use of 'copper log' and treated pine.
5. To find information on sustainable Ecosystems focusing on maintaining the sustainability of Australia's landscapes, environments, communities and the 'CSIRO's activities in sustainable agriculture.
6. To find information on Air Guitar and Textile sensor and how are they work? Additionally also gain knowledge on Biomedical and Medical textiles which might also be called as smart textiles? How does the computer translate the gestures into music? Is it wireless or wired?
7. To find information about the carbon nanotube projects at CTFT including carbon nanotube yarns (strand of twisted threads) and sheets.
8. To find information on Australia's world leading aerospace research on scramjet technology (ultra high velocity air breathing engines). NASA and others are working with Australian scientists because their own programs have been comparatively unsuccessful.
9. To find information on Human Nutrition Clinic facility and the latest clinical nutrition trials which are recruiting volunteers?
10. To find information about the dietary trial which lead to the development of Total Wellbeing Diet book and fact sheet about differing diets, etc.
11. To find information on product line lifecycle assessment including the analysis of all associated costs across entire production chain, cradle to grave analysis and assessment of long term sustainability of industries/processes.
12. To find information about comminution (fragmenting), using physical techniques such as crushing and grinding to access minerals in ore (rock) bodies.

- 13.** To familiarise yourself with improving cement production techniques such as slag granulation, use of waste heat and use of different materials to replace traditional Portland cement (e.g. geopolymers)
- 14.** To find information about Sensor networks and its applications? How does this technology work?
- 15.** To find information on telecollaboration, virtual offices, and the new technologies exist to support collaboration across distance. In addition, find out whether these technologies are safe, private, secure and reliable.
- 16.** To find information on mining robotics including what mining technologies can be automated? How this could be done? How do robots navigate? And how safe is the technology?
- 17.** To find information about drilling including the new drilling technologies, diamond composite drill bit, autonomous drilling and cutting.
- 18.** To find information on ocean currents and conditions. The required information might include ocean heat transport and circulation and the Antarctic circumpolar current that connects all of the major ocean basins to the south of Australia and the Indonesian Throughflow.
- 19.** To find information about the science education programs which 'CSIRO' operates in different school around Australia. Such programs may include awards, competitions, student workshops, research projects, teacher professional development and teacher resources.
- 20.** To find information about genome damage and its' relation to cancer risk in humans.

9.2. APPENDIX 2: THE CONSENT FORM

CONSENT FORM



Title of Project:

An Adaptive Fuzzy Based Recommender System for Enterprise Search

Name of Researcher:

OBADA Y A ALHABASHNEH

Please initial box

1. I confirm I have read and understand the provided information sheet for the above study and have had the opportunity to ask questions.

☐

2. I understand that my permission is voluntary and that I am free to withdraw at any time, without giving any reason, without my legal rights being affected.

☐

3. I agree to take part in the above study.

☐

4. I would like to receive a summary sheet of the experimental findings

☐

Name of subject

Date

Signature

Researcher

Date

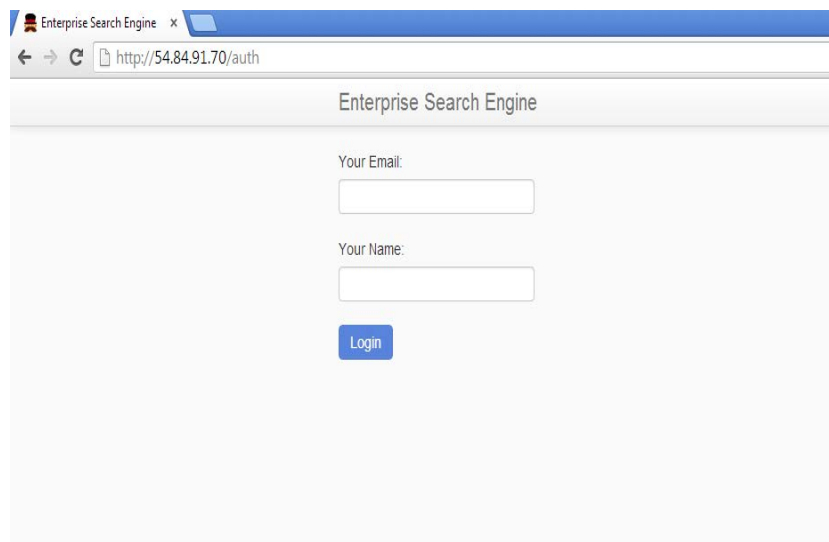
Signature

9.3. APPENIX 3: USER GUIDE

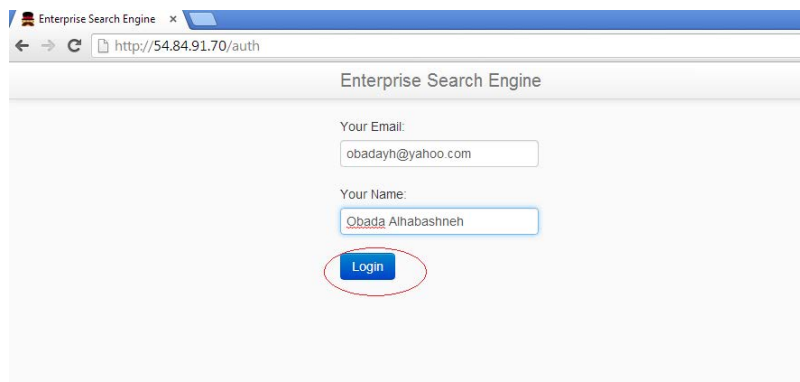
USER QUICK GUIDE

This document is a user guide to help you to use the system and to complete the provided search tasks in the right way. The experiment consists of 20 search tasks which are provided in the attached file (Search Tasks), please go through each task try to understand what is required and then use you own criteria to find the required information. The following is a step-by-step guide to show you how to use the system:

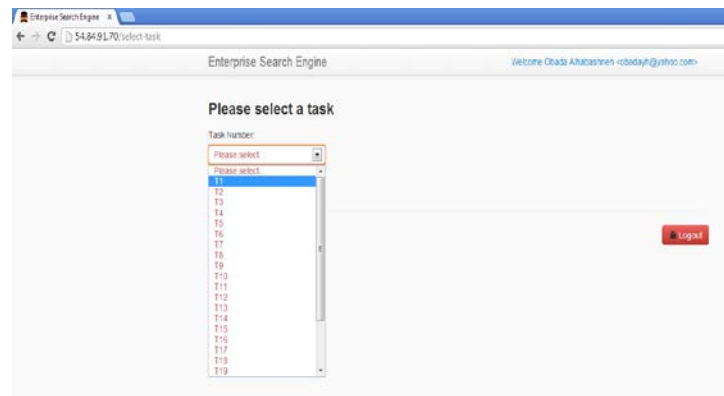
Open the system from this link: <http://54.84.91.70> and you will get this page:

A screenshot of a web browser window showing the login page of the 'Enterprise Search Engine'. The browser's address bar displays 'http://54.84.91.70/auth'. The page has a light gray background with a white header area containing the text 'Enterprise Search Engine'. Below the header, there are two input fields: 'Your Email:' and 'Your Name:'. A blue 'Login' button is positioned below the 'Your Name' field.

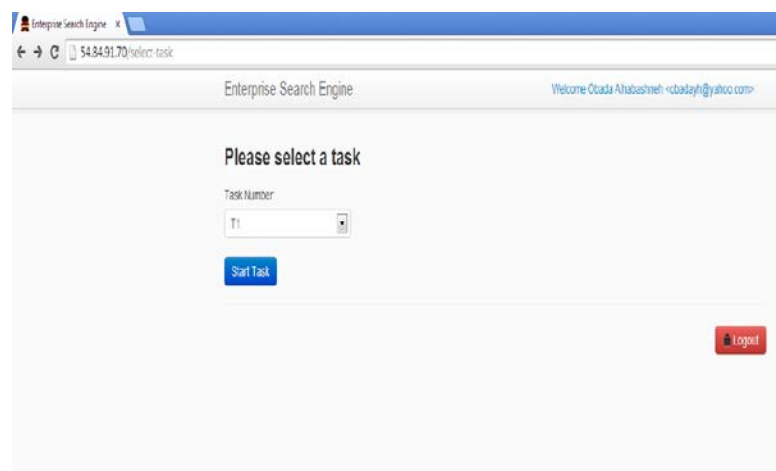
Then you need provide your email and your name (please choose make sure to use the same email and name every time you login to the system). And then click on the button 'Login'

A screenshot of the same 'Enterprise Search Engine' login page, but with the input fields filled. The 'Your Email' field contains 'obadayh@yahoo.com' and the 'Your Name' field contains 'Obada Alhabashneh'. The blue 'Login' button is now circled with a red oval, indicating it should be clicked.

Then you need select the search task that you want do from the list (you should have read the task and understood it already the attached tasks document).



Then you need click on the button 'Start Task'



Then you need search for the required information using your own queries and you can reformulate your query until you are satisfied with information you found.

